Face Liveness Detection under Processed Image Attacks

OMAR, LUMA,QASSAM,ABEDALQADER

Face Liveness Detection under Processed Image Attacks

Luma Omar

A Thesis presented for the degree of
Doctor of Philosophy

Innovative Computing Group
Department of Computer Science
University of Durham
United Kingdom

April 2018

Dedications

To my better half, who enlighten my entire life with his unwavering love. To the selfless hero who stood by my side and gave me strength I never knew I had.

To my beloved husband "***Saleh Al Assaf***"

***

To whose love, affection, encouragement and prays of day and night make me able to get such an accomplishment and honor.

To my kindhearted king, my father "***Qassam Omar***".

***

To whose the paradise beneath her feet. To the most ambitious person who was the ultimate source of motivation and inspiration.

To my role model in this life, my mother "***Mervat Khayyat***".

***

To who pushed me beyond the limits, and overwhelmed me with everlasting love and warm encouragement.

To my brothers "***Abedalqader***" & "***Mahmoud***", and sweetheart sister "***Ala'a***".

***

To the apples of my eyes, my adorable nieces, "***Jana***" & "***Talia***".

***

To my uncles, aunts, and their families. And to my friends in Jordan, United Kingdom, Qatar and all over the world.

***

To the soul of my dearest friend, who was always a great companion, "***Sarah Hannoun***", may ALLAH rest her soul in eternal piece and have HIS mercy on her.

***

To every individual who was generously supportive .. I dedicate this achievement to each one of you, with love.

Face Liveness Detection under Processed Image Attacks

Luma Omar

Submitted for the degree of Doctor of Philosophy
April 2018

Abstract

Face recognition is a mature and reliable technology for identifying people. Due to high-definition cameras and supporting devices, it is considered the fastest and the least intrusive biometric recognition modality. Nevertheless, effective spoofing attempts on face recognition systems were found to be possible. As a result, various anti-spoofing algorithms were developed to counteract these attacks. They are commonly referred in the literature a liveness detection tests. In this research we highlight the effectiveness of some simple, direct spoofing attacks, and test one of the current robust liveness detection algorithms, i.e. the logistic regression based face liveness detection from a single image, proposed by the Tan et al. [177], against malicious attacks using processed imposter images. In particular, we study experimentally the effect of common image processing operations such as sharpening and smoothing, as well as corruption with salt and pepper noise, on the face liveness detection algorithm, and we find that it is especially vulnerable against spoofing attempts using processed imposter images. We design and present a new facial database, the Durham Face Database, which is the first, to the best of our knowledge, to have client, imposter as well as processed imposter images. Finally, we evaluate our claim on the effectiveness of proposed imposter image attacks using transfer learning on Convolutional Neural Networks. We verify that such attacks are more difficult to detect even when using high-end, expensive machine learning techniques.

Declaration

The work in this thesis is based on research carried out at the Innovative Computing Group (ICG), the Department of Computer Science, University of Durham, United Kingdom. No part of this thesis has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text.

Acknowledgements

ALHAMDULILLAH, I wholeheartedly praise and thank ALLAH (SWT), the Almighty, for HIS uncountable blessings, giving me the strength and power to carry on this research, letting me live and see my thesis go through, and making my dream come true.

This thesis is a culmination of my journey towards the Ph.D. While my name may be alone on the front cover of this work, I am by no means its sole contributor. Rather, there are a number of people behind this piece of work who deserve to be both acknowledged and thanked here.

First and foremost, I am very fortunate and grateful to my supervisor, Dr. *Ioannis Ivrissimtzis*, who has meticulously supported me throughout my study with his patience, continuous guidance, knowledge, suggestions and astute criticism.

I am extremely grateful to my husband *Saleh*, my beloved parents, brothers *Abedalqader* & *Mahmoud*, sister *Ala'a*, and nieces *Jana* & *Talia*, who have been a constant source of love, inspiration and encouragement during the challenges of the graduate school and life.

My sincere gratitude to the rest of my family and friends everywhere, who were always encouraging me in all of my pursuits.

I specially acknowledge Mr. *Hussein Arouri* and Mr. *Mahmoud Alawamleh*. I convey my deep appreciation and respect for both of you, without whom my ambition could not be accomplished.

This journey would not have been possible without the support from each of you.

Publications

The contents of this thesis are based on the results from the following papers.

- Omar, L., and Ivrissimtzis, I. Evaluating the resilience of face recognition systems against malicious attacks. In *BMVW* (2015), pp. 5.1-5.9. (Chapter **4**)

- Omar, L., and Ivrissimtzis, I. Resilience of luminance based liveness tests under attacks with processed imposter images. In *WSCG* (2016), pp. 79-82. (Chapter **5**)

- Omar, L., and Ivrissimtzis, I. Designing a facial spoofing database for processed image attacks. In *ICDP* (2016), IET, pp. 5(6.) - 5(6.)(1). (Chapter **6**)

In addition to the above, the following paper was produced during my Ph.D studies and it is still unpublished.

- Omar, L., and Ivrissimtzis, I. Signal-Noise Analysis of Face Anti-spoofing under Processed Image Attacks. (Chapter **7**)

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Biometrics based user verification systems rely on the extraction of some human biological characteristics and their statistical analysis to verify the identity of a person. Biometric Security is a relatively new technology which has seen its usage rapidly in the last few decades. It can compete against the more traditional methods such as pins and passwords. which can be easily guessed, or forgotten, leaving the person to struggle with no access to the system [50, 81].

A biometric characteristic is any extracted measurable distinguishing characteristic of an individual used for the purpose of biometric identification. Biometric characteristics are classified into two main categories; physiological and behavioural biometrics. Physiological biometrics are based on human body part measurements and prominent examples include fingerprint, face and iris recognition. Meanwhile, behavioural biometrics are those based on human action measurements and prominent examples include gesture, key stroking, gait and signature recognition [16, 50]. For any of these characteristics to be qualified as a biometric, it should satisfy the following requirements:

- Universality: where each individual should have the determined characteristic.
- Distinctiveness: no two individuals must have the same characteristic.
- Permanence: the characteristic should be permanent and invariant over a duration of time.

- Collectability: there should be a quantitative measurement for the characteristic.

  Moreover, practical biometric systems should also consider the following issues:

- Performance: resources require both intended recognition beside the operational and environmental factors to achieve the desired accuracy and speed.

- Acceptability: users are willing to accept the use of biometric identifier in their daily lives.

- and Circumvention: indicates the easiness or difficulty of fraudulent attempts on the system.

Among various physiological biometric methods, face recognition has recently received attention from both industrial and academic fields [209]. Currently, face recognition is one of the most widely used authentication methods based on biometrics. It is considered as a mature technology which offers a fast, reliable, convenient and inexpensive way for person identification and has already found a wide range of applications, from security critical applications such as passport control at the gates of an airport, to consumer level applications, such as automatically logging into a laptop or smartphone. The developed techniques vary in sophistication, as well as in hardware and software requirements, ranging from systems based on 3D face scans, through systems based on videos, to systems that can work with a single still image [143]. In contrast to the fingerprint and iris recognition which use high resolution images, face recognition has the unique characteristic of being based on data that can well be found in the public domain. Regarding performance and acceptability, it is a prime candidate technique in biometric identification applications requiring real-time, reliable and unobtrusive user authentication without the use of specialized hardware.

Regarding circumvention, face recognition is considered vulnerable to spoofing attacks. This vulnerability means that user authentication through face recognition is still mostly confined to either applications with low security requirements, or applications in highly controlled environments.

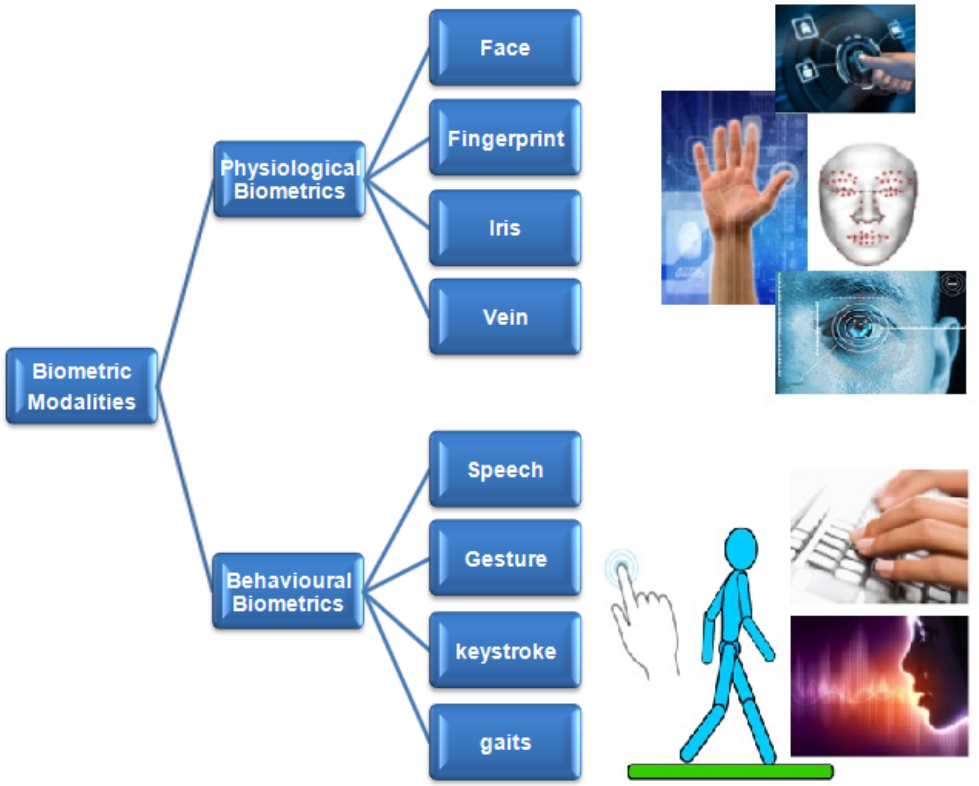


Figure 1.1: Biomteric Modalities; Physiological and Behavioural biometric methods.

Enhancing the security of face recognition systems is a major challenge since a secure system should be able to withstand a variety of attacks, ranging from systematic algorithmic attacks to attacks based on theft of data. In one of the simplest spoofing attacks on a face recognition system, the relatively easy access to face image data of the average person, allows a spoof attack by printing on paper the photo of the rightful user of the face recognition system and hold it in front of the camera to gain access to the system. These photos can be easily obtained either by doing a quick on-line search or by logging into a social network and downloading photos or videos they have posted on these sites and hold them in front of the camera. This means that face recognition is particularly vulnerable to spoofing attacks.

As a result, the use of face recognition is restricted to either applications where security is considered secondary to convenience, e.g. log in to a personal devices, or to applications in tightly controlled environments, such as issuing national ID cards,

where the behaviour of authentication systems is closely observed. When either of these two conditions are not met, e.g. money withdrawals from a street ATM machine, face recognition is not deemed a suitable person authentication method.

As a response to such vulnerabilities, the development of anti-spoofing algorithms and techniques, commonly called *liveness tests*, has become a very active research area. Liveness tests are binary classification algorithms aiming at determining whether the recognised face is a live face, or for example, a photo or video played in front of the systems camera by an attacker.

The performance of anti-spoofing algorithms is evaluated on databases containing both photos of real people called *client* images, and spoofing photos, which essentially are photos of client images and are called *imposter* images. The design of such a database is a particularly challenging task given the multiple sources of variation in spoofing attacks. Indeed, a whole range of choices, from the choice between a paper photo or an electronic display for the attack, to the type of paper and printer used to print a photo, to the size of that photo and the way it is held in front of the camera, all these factors can impact the effectiveness of the attack and thus the perceived performance of the anti-spoofing algorithm.

1.1 Motivation

Over the years, a large number of face anti-spoofing algorithms have been developed. The variety of these approaches and the research interest in the area, motivated us to assess some of the anti-spoofing algorithms against attacks.

Due to the numerous potential face spoofing attacks, we decided to first evaluate the resilience of some well-known face recognition systems against a pre-designed, direct and simple attack. The aim was to emphasise the easiness of spoofing some widely available face recognition systems. Our experiment showed that some of these widely used applications are strongly vulnerable against various types of direct spoofing attempts, see Chapter 4.

Chapter 4 tested hardware and software systems rather than algorithms. In our next step we moved from black box testing to white box testing, concentrating on

some fundamental anti-spoofing techniques.

Tan et al. [177] is a well-known, robust face liveness detection technique, see Section 3.4 for details. It is a relatively simple algorithm and all its components are based on well-understood techniques. In Chapter 5 we chose to test this algorithm by processing an imposter image with different image processing techniques and also applying certain filters to these images, such as the noise addition, sharpening, and blurring.

In particular, latent samples were constructed using the difference of Gaussian (DoG), because the Fourier spectrum of the DoG for both client and imposter images shows that the real image has a richer horizontal components in the high frequency areas than the imposter image. In Figure 1.2, Tan et al. is visualizing the 2D Fourier transform of the DoG for both client and imposter images from their pre-collected NUAA facial spoofing database. In our research, we are studying the robustness of the Tan et al. proposed face anti-spoofing algorithm using processed imposter images under various filters. We mostly use three common filters; sharpening, adding salt and pepper noise, and Gaussian blurring. The aim is to identify the most suitable filtering technique which increases the horizontal frequency areas of the imposter image to become more similar to client images, and thus, it makes it more difficult to distinguish between them.

As an example to our intended work, Figure 1.3 shows the 2D Fourier transforms of DoG of images taken by us for testing the effectiveness of spoofing attacks and the result shows differences in the frequencies around the central areas. Figure 1.2 (f) shows how the horizontal components in the high frequency areas of the sharpened imposter image became richer.

Our initial investigations highlighted one limitation shared by the majority of the current approaches to the development of liveness tests, that is, the tacit assumption that the imposter will use the stolen image or video as it is, i.e., without previously processing it in order to increase the effectiveness of the attack. Moreover, this tacit assumption is carried over from the development to the evaluation of liveness tests. Thus, the most popular image and video databases for evaluating liveness tests, such as the NUAA [177], PRINT-ATTACK [14], REPLAY-ATTACK [33]

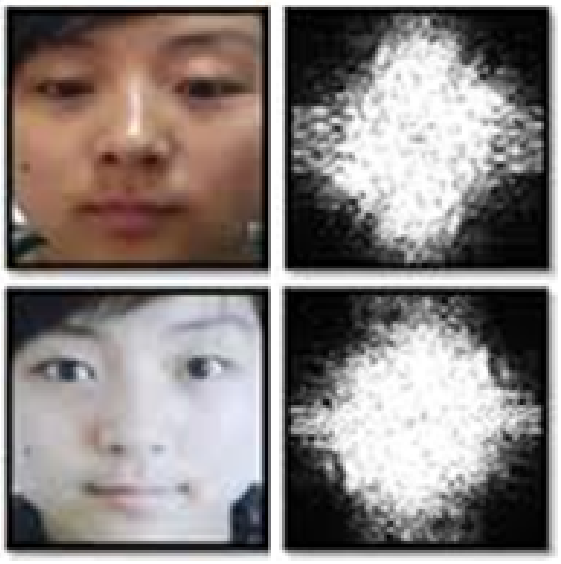


Figure 1.2: **Top-left:** the real image. **Bottom-left:** the imposter image. To the right of each image is the 2D Fourier transform of the DoG image [177].

and CASIA [33] consist of client and imposter images, or videos captured from unprocessed photographs or videos of the users. In contrast, the DURHAM FACE (DF) database [131], developed as part of this project, contains both usual imposter images, imposter images obtained by photo-shooting printouts of sharpened client images, and processed images displayed on a digital displays. On the other hand, it is considerably smaller than the previous ones.

For creating a robust face spoofing database, a wide variety of parameters should be considered, e.g. the camera used, the camera focus mechanism, the object distance and the type of printer that was used. Most publicly available databases do not consider a wide range of parameters, so we intend to experiment with a set of potentially useful parameters before the creation of the database. For example the performance of a liveness test can be affected when using different printers to create printed imposters. Besides, the problem becomes even more challenging when considering imposters being taken from different distances. See Figure 1.4.

1.2 Research Questions

In this thesis we aim to answer the following questions:

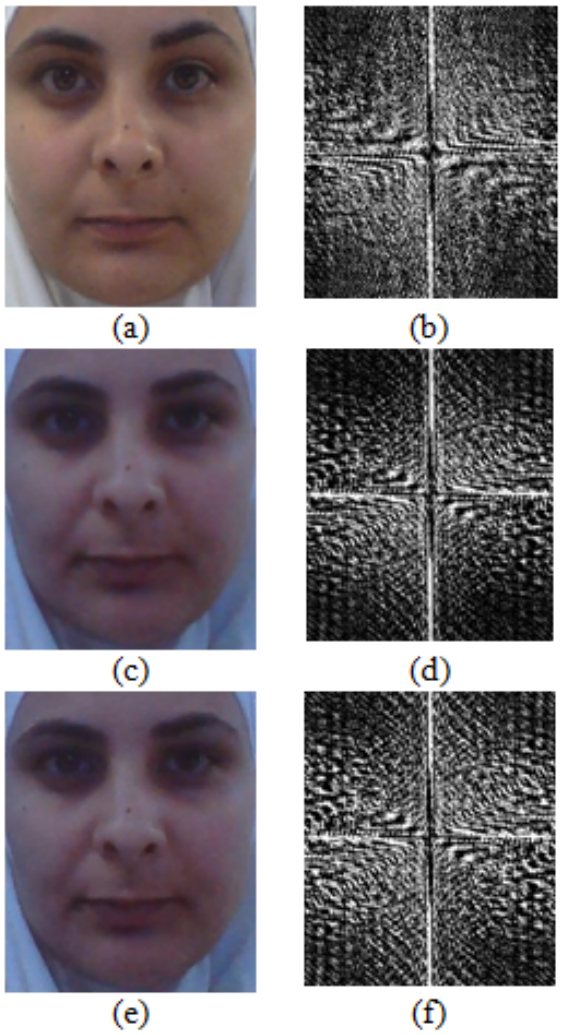


Figure 1.3: (a) the real image (b) the 2D Fourier of the DOG filtered real image (c) the imposter image (d) the 2D Fourier of the DOG filtered real image (e) Sharpening filter added to the imposter image (f) the 2D Fourier of the DOG filtered sharpen imposter image.

1. Regarding commercial face recognition softwares and applications.

   (a) To what extent are current commercial face recognition softwares robust against even crude spoofing attacks?

   (b) Does the performance of liveness detection in face recognition systems affected by using a processed images i.e., noise addition, or blurring?

   (c) Can we identify the compression threshold above which a photo can be used to gain access to a face recognition system.

Figure 1.4: Imposters with different distances from camera and printers quality, respectively. **(i)-(ii)** Imposter was shot from: (i) 6 cm, (ii) 9 cm distance from camera, **(iii)-(iv)** Imposters was printed using two different printers (iii) Ricoh 4500, and (iv) Bizhub c654e.

2. The robustness of available facial anti-spoofing algorithms.
   (a) To what level would the liveness detection algorithms counterfeit spoofing attempts?
   (b) Does processing images in general before using as imposters change the performance of a liveness test?
   (c) How does each of sharpening, blurring, and noise addition affect the robustness of anti-spoofing technique? and which values are the best/worst?
3. Facial spoofing databases
   (a) What are the current publicly available face spoofing datasets? and do these datasets fulfill researchers requirements?
   (b) Can we create processed imposters facial spoofing database?
   (c) To what extent can we define parameters for the best database creation?
4. Convolutional Neural Networks
   (a) Can the use of deep learning techniques increase the robustness of face recognition systems against spoofing attacks with processed imposter images?

1.3 Contributions and Limitations

The first part of our research corresponding to Chapter 4 of the thesis. We present an experiment designed to evaluate four of the well-known commercial face verification systems against direct spoofing techniques. The test has two parts; the first part uses still face images collected in different ways and from various sources, i.e., trying to gain access to the system by using a mobile photo of people, their IDs or photos based on on-line social media means. In the second part of the experiment, we resize some of the still images that were successfully been used to gain access to the system and we find the minimum resolution required for such an attack. Although all of the tested software was found spoof-able, we believe that some companies set the default configuration to prioritise user convenience over security which makes the evaluation of the potential resilience of this software against various attacks difficult to measure.

The next part of our research corresponding to Chapter 5 of the thesis, testing the performance of Tan et al. [177] algorithm against the claim that processing operations applied to imposter images, like sharpening and smoothing, can be successful in attacking face recognition systems. The main limitation of this part is that we process images from NUAA database without these images being physically recaptured. Although this approach allows for a better understanding of the basic principle of the Tan et al. [177] and provides a clear idea of the effect of processing images on detecting the liveness of the object in front of the face recognition system, the real effect of the direct attack using processed images is still unquantifiable without feeding the algorithm with physically recaptured images.

In the next step, corresponding to Chapter 6 of the thesis, we recaptured processed imposter images, and we introduced our DURHAM FACE (DF) database. DF database has been designed and constructed as a dataset of face images for testing anti-spoofing techniques. DF database is, to the best of our knowledge, the first database based on the assumption that the attacker may use image processing tools to enhance the effectiveness of their attack. Our current database only serves as a proof of concept, considering only one image processing operation on the client images before they are printed. However, extending the database with imposter im-

ages that have undergone other types of processing is a relatively straightforward, even though laborious, process.

The next step, corresponding to Chapter 7 of the thesis is to explore statistical analysis techniques that go beyond the drawing of the ROC curves, or simples related measures such as AUC (Area Under the Curve) which have some well-known limitations [66]. Inspired by [67], we used the signal-noise decomposition model based on beta distributions for studying the behaviour of liveness tests under processed image attacks with the amount of sharpening treated as a parameter. We evaluate this model with the fact that both with a classic liveness test and with a tailor-made one, aiming at exhibiting the variety encountered in the behaviour of different liveness tests under simple image processing operations. We tested the database on a standard liveness test [177] and found that the more sophisticated attack with processed imposter images is more likely to evade detection. The main limitation of this part is that following, our previous work in Chapter 5, we evaluate our approach indirectly, that is by direct processing of the imposter images, instead of processing client images, printing them and taking photos of them which are then used to produce imposter images. However, we note that the validity of this indirect approach has already been verified in the results of chapters. Moreover, it is again based on the very reasonable assumption that if a digital image is sharper than another, then it will most likely remain the sharper one after both images are printed on paper and recaptured on a camera.

Convolutional Neural Network (CNN) is one of the deep neural network models applied on visual imagery problems. Despite the lack of large scale dataset, with let say hundreds of thousands of images, we are still able to use deep learning to validate our assumption that using processed imposters can enhance an attack on face recognition systems. In the last part of our research corresponding to Chapter 8, we are using transfer learning in CNN to evaluate the performance of face recognition systems against malicious attacks using processed imposter images. A very commonly used pre-trained CNN, the VGG-16, was employed to evaluate our claim, using the extended DURHAM FACE database, which consists of real face images, unprocessed and processed by various amount of sharpening imposters, created by

being played on a digital display and recaptured by a professional camera.

The main contributions of our work can be summerised as follows:

- We are studying and demonstrating the deficiencies of some well-known commercial face recognition systems against direct spoofing attempts, i.e. using recaptured images of clients like ID photos or images on social media. This study is cited in [130].
- Testing the performance of one of the well-known algorithms proposed by Tan et al. in [177] against processed images, i.e. sharpened, smoothed, and images with added noise. This study is cited in [132]
- Introducing our own face spoofing database, the DURHAM FACE (DF) database, which contains a set of real, imposter, and processed imposter images, for evaluating spoofing techniques. This work is cited in [131].
- Testing the behaviour of using a tailor made shallow neural network against malicious processed images attacks by the signal-noise decomposition model based on beta distributions.
- Proposing the use of the transfer learning in convolutional neural network, the VGG-16, to study the performance of face liveness detection using our own dataset, the DURHAM FACE database.

1.3.1 Evaluation Methodology

Face liveness detection algorithms were found to be spoofable, we are evaluating the robustness of some of these anti-spoofing techniques against malicious attacks, including using processed imposters such as sharpening, blurring, and noise addition. We are evaluating our work in this research with several mathematical techniques as follows:

- **ROC curve.** We use the Receiver Operating Characteristic to plot the True Positive Rate (TPR) against the False Positive Rates (FPR) at different thresholds. This method represents the sensitivity or the probability of detection in Machine learning cases.

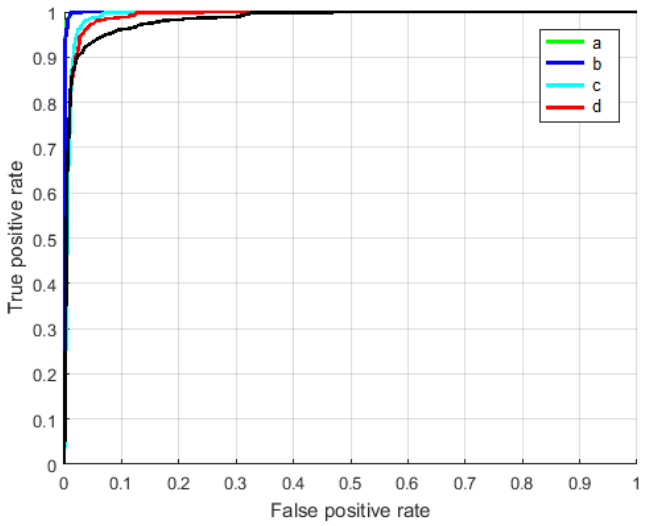


Figure 1.5: An example of ROC curves.

ROC is the most common method to evaluate the performance of the classifier. In some of our cases we noticed that curves intersects and the whole family of ROC curves pass from a single point. So for example see Figure 1.6. While there is nothing unusual in this behaviour, it might lead to a misleading interpretation of the classifier's performance. That inspired us to study the signal-noise decomposition of the output in Chapter 7.

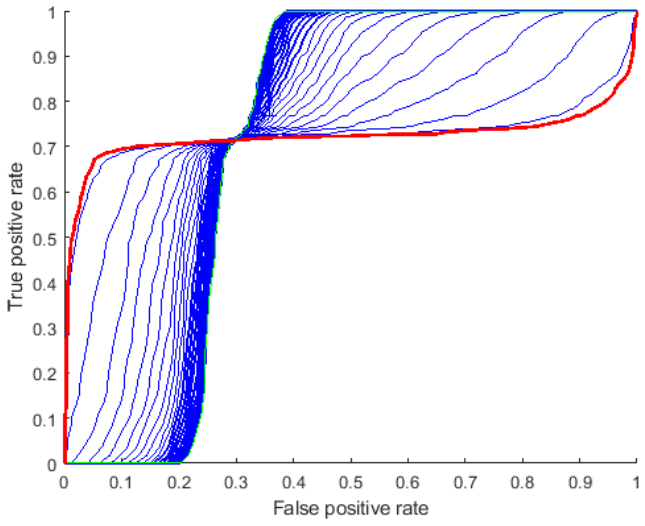


Figure 1.6: An example of ROC curves passing from a single point.

- **t-test.** is a statistical comparison of two populations, and used when the

variance of two normal distributions are not known. In our research, we use the t-test to inform the design of the DF database produced by various types of cameras. There were many choices for creating imposter, e.g. type of camera, focus mechanism, etc. and we run small pilot experiments to see when some of them are statistically different from others. Further implementation of the *t-test* in section 6.3.

More details of evaluation methodologies can be found in the relevant chapters.

1.4 Thesis Overview

Our thesis is structured as follows. In the next two chapters, Chapter 2 and Chapter 3, we review different biometric modalities, face, fingerprint, iris, gait and more. Afterward, we discuss various spoofing techniques on different biometric systems and we provide a review of a wide range of anti-spoofing approaches. We review face liveness detection and some well-known and widely used facial spoofing databases. In Chapter 4, we present an experiment designed to test the resilience of face recognition systems against malicious theft of data attacks. In Chapter 5, we explore potential vulnerabilities of the liveness test proposed in Tan et al. [177] by studying the effect on its performance of simple image processing operations applied on the imposter images, such as sharpening and smoothing. We introduce our own database (DURHAM FACE database) in Chapter 6 which contains imposter images obtained by photo-shooting printouts of sharpened client images. Chapter 7 aims at a better understanding of the behaviour of liveness tests against processed imposter image attacks, employing a more detailed statistical modelling of the output of the classifiers, beyond the plotting of empirical ROC curves. In Chapter 8, we present the use of the VGG-16 pre-trained Convolutional Neural Network in assessing the performance of face recognition systems against processed imposter image attacks. We conclude and summerise our work in Chapter 9 and discuss the intended future plans in that last section.

Chapter 2

Literature Survey

This chapter will provide a detailed background on various biometric recognition systems of various types. We focus on reviewing face recognition systems but also cover other biometric traits i.e., fingerprint, iris, vein, and gait, along with multi-modal biometrics. Moreover, this chapter will provide a survey on biometric systems spoofing techniques in both direct and indirect form, for various modalities.

2.1 Biometric Traits

Biometric recognition is based on the ability to uniquely identify a person by extracting one or more distinguishing biological traits and processed with their statistical analysis. These biological characteristics can be a person's face, fingerprints, retina and iris patterns, hand geometry, voice, DNA, or hand-written signatures.

A biometric system has four main modules; sensor, feature extraction, matcher, and system database. It performs three main tasks; enrollment, verification, and identification. *Enrollment* is responsible for enrolling users into the system's database by recording the user's biometric trait using an appropriate *sensor* i.e., a camera for face or a scanner for fingerprint inputs. Then, salient characteristics are being extracted using an algorithm called *feature extraction*. Next, these extracted features along with pre-defined identifiers such as names or numbers are being stored as a *template* in the database. For authentication process, the user provides the system with a *query*; a different sample presented in front of the sensor, which then

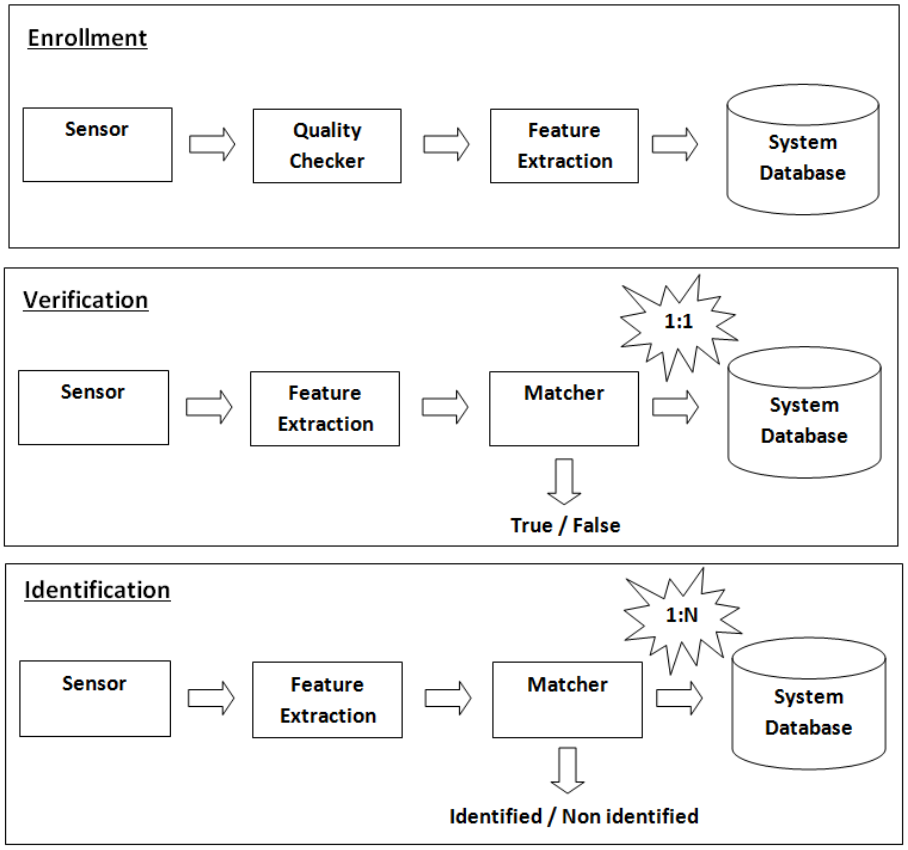


Figure 2.1: Enrollment, verification, and identification tasks of a biometric recognition system.

is being compared to the pre-stored template via the *matcher* that returns a match score to indicate the similarity between the template and the query. A match score should be higher than a predefined threshold for the system to accept the identity claim [82]. Figure 2.1 illustrates how does a biometric system operate.

Biometric systems have also been defined as pattern recognition systems which compare the collected biometrics from individuals after feature extraction with the pre-stored user feature data in the database. Biometric systems can be operated in either *verification* or *identification* modes. *Verification* mode is used for positive recognition, where a one-to-one comparison process takes a place between the data provided by the individual and data stored in the database, and aims to prevent different people gaining access to the same identity account. In general, a verification task validates the identity of the user by comparing the captured biometric to what is already stored in the database of the system. Meanwhile, *identification* mode is

used for Negative recognition, where the database is being searched for a match to the provided template by the user. Overall, identification tasks can be summarised as a recognition of the user occurring by searching the template of all individuals in the database for a match with a one-to-many comparison [82].

When verification errors occur in biometric systems, these errors are one of two types: False Matching Error (FMR) where some imposter can be accepted by the system, and False Nonmatching Error (FNMR) where some valid individual can be accepted by the system [82].

Biometrics are considered more reliable than other access mechanism since they cannot be stolen, lost, duplicated or left behind at home or any other place like physical access items (cards, keys, tokens, etc.). Nonphysical access items (passkeys, PINs, passwords, pass patterns, etc.) can be easily forgotten, stolen, observed, or even shared between more than one people. In contrast, these problems do not appear in biometric systems [196].

Biometrics are comprised of biological behaviour and behaviour traits [38]. Biological behaviour which is also known as physiological biometric methods are based on human body part measurements and prominent examples include face, fingerprint, and iris recognition. Behavioural biometric methods are based on human action measurements and some well-known examples are gesture, key stroking, gait and signature recognition [16,50]. Gesture recognition identifies the person by states generated from any body motion, but commonly originate from either the face or hand. Meanwhile, keystroke recognition is a technique of recognizing the user by the rhythm he/she uses while typing characters using a keypad or keyboard. Another common behavioural biometric is gait recognition, where the individual is been identified from the way he/she walks, as it has been found that each person has a unique way of walking and can be used for recognition purposes. Signature recognition is also another behavioural biometric where the user authentication occurs using the signature and can be operated in two ways; static or dynamic. Static where the user insert the signature using any mean of inputing and then the system recognizes the inserted template. In contrast, dynamic signature recognition recognizes the person from his time-strokes of signature on digital displays, in real time.

2.1.1 Face Recognition

Face recognition is a well established research area with the state of the art techniques achieving recognition rates that rival the human ability to recognize faces under similar conditions. The input of a face recognition algorithm can be a grayscale or a color still image, a short video sequence or a 3D scan of someones face. The first examples of successful face recognition algorithms from still images were based on Principal Component Analysis (PCA) [163] and [180, 181], while further improvements proposed in [142] were able to cope with large scale databases and handle better the problem of pose variability by using modular eigenspaces. In Barlett et al. [18], Independent Component Analysis (ICA) was employed instead of PCA.

Compared to the other main biometric authentication methods that use fingerprints or high resolution iris images, face recognition has the unique characteristic that it is based on data that can well be found in the public domain. Indeed, in many cases it is very easy to obtain a photo of someone's face, either doing a quick online search or by logging into a social network. As a result, face recognition based authentication is particularly vulnerable to imposter attacks, when, for example, an attacker holds someone's photo in front of the camera and try to gain access through a face recognition system.

In [209] and [197], face recognition methods are classified into three main categories: (1) Holistic matching methods, (2) feature-based (structural) matching methods, and (3) hybrid methods. Holistic methods use the whole face region as input of the face recognition algorithm. Eigenfaces [181], Fisherfaces [21], and Laplacianfaces [72] are the common methods of extracting holistic face features. Principal Component Analysis gives the best known example of Eigenfaces. The basic algorithm was proposed by Sirovich and Kirby [163] and used for face classification by Turk and Pentland [180]. The eigenfaces are the eigenvectors with the lowest eigenvalues of the covariance matrix of a high dimensional vector space of face images. They are used as the basis of a lower dimension vector space of face images. The reduction in the dimension allows for an efficient solution to the face classification problem. Figure 2.2 presents a sample of eigenfaces calculates from sample images from the DURHAM FACE database, see Chapter 6. Furthermore, the figure shows

the normalized faces and the mean face which has been used to compute eigenfaces.

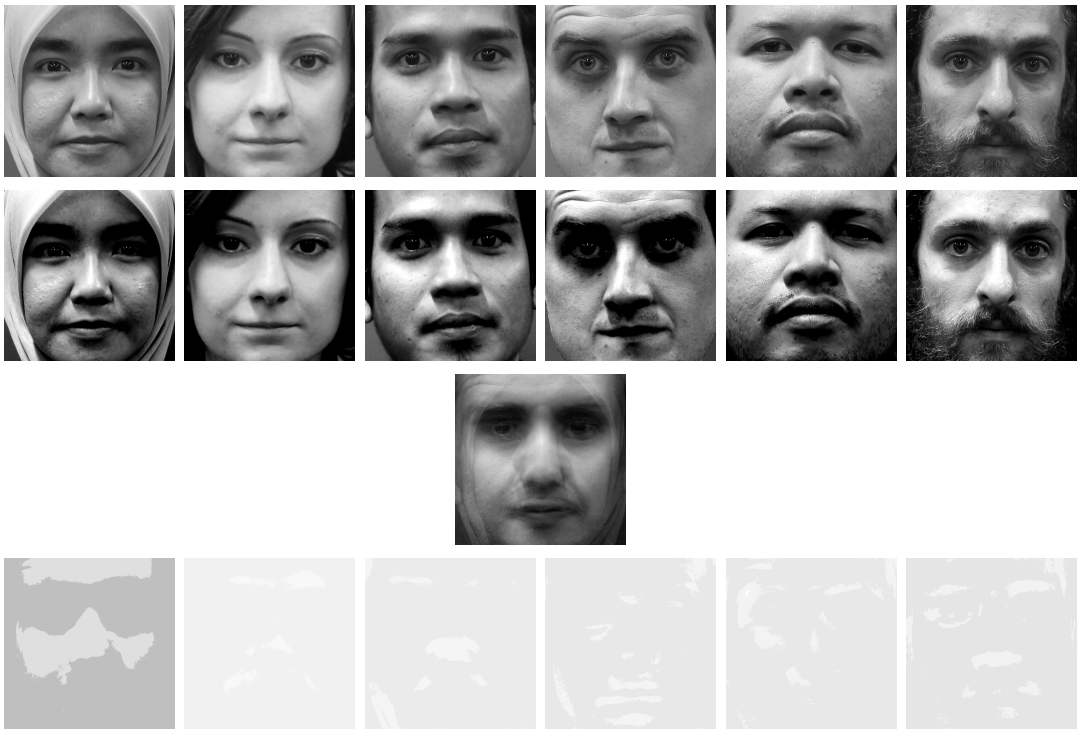


Figure 2.2: Samples of Eigenfaces, calculated from a sample of images from the Durham face database. **Row 1**: Grayscale sample images from DURHAM FACE database. **Row 2**: normalized faces. **Row 3**: the mean face. **Row 4**: Eigenfaces.

Fisherface face recognition method uses Linear Discrimant Analysis (LDA) to extract discriminative features of the face, instead of using Principal Component Analysis (PCA) to find a subspace representation of a set of face images. The Fisherface technique was found to have lower error rates compared to the Eigenface technique [21].

Meanwhile, LaplacianFace is another linear method for face recognition which models the space of faces as a manifold structure. It uses the Locality Preserving Projections (LPP) to map the face images into a low dimensional face space. While Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) effectively assume a Euclidean structure for face space, the LPP in contrast finds optimal linear approximations to the eigenfunctions of the Laplace Beltrami operator on a manifold. LaplacianFace gives lower error rates and a better performance compared to the two previous methods [72].

In feature-based methods, local features such as eyes, the curve of eyebrows, nose, mouth, and the shape of the lips and chin, are first extracted and their locations and local statistics (geometric and/or appearance related) are fed into a structural classifier [209].

Hybrid methods combine both feature-based methods and holistic methods. For example, [142] combines Eigenfaces with Eigenmodules (Eigenmouth, Eigeneyes, Eigennose).

Other application domains of face recognition use unconstrained face images. Such images can be captured either using mobile devices or surveillance cameras [23] or images that can be found on the Internet [77]. Although the setting is quite different, the techniques developed for unconstrained face recognition are relevant to biometric security. The Labeled Faces in the Wild (LFW) dataset is one of the most recent and familiar databases for unconstrained face images, which contains 13,000 face images collected from the web. LFW labeled each image with the name of the pictured individual without any constraints been applied to the images. The only constraint in the LFW dataset is that all images had been detected by the Viola-Jones face detector. Figure 2.3 shows a number of images from the LFW. These unconstrained face images are difficult in recognition due to the poor image quality, inconsistent poses, expressions and orientations, variation in brightness [47].

Figure 2.3: Sample images from the LFW dataset. The first row shows random samples of different subjects, while the second row shows different images of the same subject.

Lately, using both "Shallow" and "Deep" neural networks for face recognition in images and videos has received considerable attention. At the beginning of using shallow methods in face recognition, handcrafted local image descriptors like SIFT, LBP, and HOG [36, 114, 164, 165, 195] where used to extract a representation of a face image.

Recently, Convolutional Neural Networks have been used to improve various classification problems; object [69,70,99,161,175], scene [210,211], and action classification [15, 85, 188]. Large scale research has been conducted to solve face recognition problems using deep learning. One of the leading researches in the area is the one by Taigman et al. [176] who trained a nine-layer deep neural network with a large face dataset of 4,000 identities. Some other leading researches in face recognition use deep learning; Chopra et al. [34] presented a method to be used for recognition or verification applications by learning a similarity metric from the data. Face recognition using unsupervised learning was proposed by Huang et al. in their work [76]. Sun et al. [172] proposed the use of hybrid Convolutional Network (ConvNet)-Restricted Boltzmann Machine (RBM) model for face recognition. Furthermore Sun et al. introduced two new approaches based on the Deep IDentification-verification features (DeepID) [171,173]. The face identity-preserving (FIP) features have been proposed by Zhu et al. [214].

Further researches on face recognition using deep learning has been done by Hu et al. in [74], where they proposed a new approach for measuring the performance of a face verification on the widely used faces datasets, such as the LFW and the youTube Faces database (YTF) by Wolf et al. in [195], by a new discriminative deep metric learning (DDML) method. Liu et at [113] trained a novel two CNNs systems for attribute prediction in the wild; LNet and ANet. LNet is a pretrained neural network with numerous object categories for face localization, but while ANet was trained for attribute prediction.

One challenge for the success of these methods is the large required size of the training datasets. ImageNet is an image database suitable [44] for visual object recognition experiments. Later on, the ImageNet project organized the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) which runs annually to present

datasets for general image classification tasks [153]. Google has trained the currently largest face recognition system using 200 million images of 8 million different objects [157]. This deep convolutional network is called FaceNet, and achieved an accuracy rate of 99.63% on the LFW and 95.12% on the YTF.

Video-Based Face Recognition

With the rapid development of image acquisition technology, image sequences from short videos have become a technically viable alternative to still images for face recognition because of the inability to utilize temporal information of people faces to facilitate the face recognition procedure. As a result, a multitude of techniques have been proposed to overcome challenges associated with low cost video cameras, such as low resolution and poor image quality. Techniques for face recognition from video may be based on still image face recognition methods, or on multimodal methods, combining for example video and audio, or spatio-temporal methods, analyzing for example the trajectory of face features [209]. Figure 2.4 shows samples from MoBo database along with cropped faces. The MoBo database contains video sets for 25 individuals captured using six high resolution cameras distributed around the scene [64].

In a very successful approach to face recognition from video, Liu and Chen [112] applied the adaptive Hidden Marcov Model. HMM was developed as a statistical model used to characterize the statistical properties of a signal and was previously used in modeling temporal information on applications such as gesture, expression, and speech recognition. The adaptive HMM was used to analyze the temporal characteristics of the test video sequences over time and the scores provided by the HMMs compared and the identity of the test video sequence with the highest score. In [112], the authors applied the adaptive HMM temporally to perform video-based face recognition. They assumed that each frame in the video is a unique observation and PCA was used to extract the a dimensional feature vector. Corresponding feature vectors which were used as the observation vector for the HMM training were extracted from the eigenspace. The HMM learned the statistics of the training video sequences and their temporal dynamics during the training process, while the temporal characteristics of the test video sequence were analyzed by the HMM in

Figure 2.4: Samples from MoBo database along with cropped faces [64].

real time during the recognition process. Figure 2.5, shows the temporal HMM of the Liu et al. technique for modeling face sequences.

Another research in the area proposed by Lee et al. [104] uses low dimensional appearance manifolds, a technique that was found capable in handling pose variability. An affine plane or each pose manifold, and exemplars are sampled from video and clustered with K-means algorithm. Next, Principal Component Analysis (PCA) was used to approximate each cluster by a single plane leading to a low dimensional linear subspace approximation. After that, the connectivity among these linear subspaces was represented in the format of a transition matrix. The elements of this transition matrix capture the likelihood by which frames will be making a

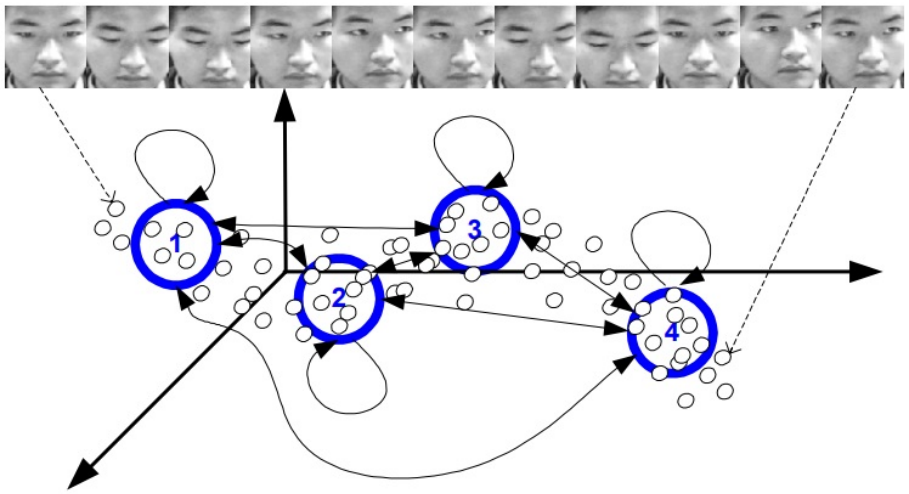


Figure 2.5: [The temporal HMM of the Liu et al. [112] proposed technique for modeling face sequences.

transition between a pair of pose subspaces.

Another video based approach is the Auto Regressive and Moving Average (ARMA) framework proposed in [8] for video-based face recognition. It was found useful for face recognition by utilizing the tentative information from the face dynamicity.

2.1.2 Fingerprint Recognition

Fingerprint is one of the oldest known biometric identification traits. It is easily available and can be considered as unchangeable over the time. Fingerprint recognition is an extremely popular and reliable verification method because of its propertied of accuracy, uniqueness, universality, and low-cost due to the availability of inexpensive scanners. It has been proven by archaeological findings that since 6000 - 7000 BC, ancients Assyrians and Chinese have used fingerprints as a method for identification [120].

Each finger of any human being has a unique mixture of ridges and valleys which forms the fingerprint [120]; ridges are lines which form the shape of a fingerprint, while valleys are the spaces between these ridges. Usually ridges and valleys run in parallel. The first forensic use of fingerprints found in a crime scene happened in Argentina back in 1893 to convict a suspect [68]. Rao [146] and Moayer and Fu [127]

were the first to implement a tree model approach in fingerprint pattern recognition, based on the ridges detection method investigated in [185].

Fingerprint features are grouped into local and global. Local features also called "minutia points", which correspond to unique characteristics of ridges such as:

- Pattern area, the part of a fingerprint that have all global features.
- Core point, the U-turn in a ridge pattern.
- Delta, the Y shaped ridge meeting.
- Type lines, two innermost ridges that are parallel.
- Ridge count, number of ridges between a delta and a core.

Meanwhile, global features are these characteristics noticeable by naked eye and form specific shapes like:

- Arch, where an arc is being formed when the ridge comes from one side and rise in the centre and then exit from the other side.
- Loop, when the ridge comes from one side and exit from the same side forming a curve.
- Whorl, where ridges form circles around the centre.

A fingerprint recognition system comprised of five stages: (i) Fingerprint Enrollment, where the image of fingerprint is being captured by the sensor; (ii) Fingerprint Image Enhancement, a processing phase to remove the possible noise inserted at the capturing level, basically in this phase a form of normalization occurs; (iii) Minutiae Extraction, where both local and global features are being extracted, these features are saved in the database for the next matching process; (iv) Minutiae Matching, here matching similarities are identified between the test templates and the master template. For a one-to-one matching process, or otherwise one-to-many for verification; and (v) Fingerprint Authentication to return the result of the classification.

Fingerprint recognition techniques can be classified as Minutiae-based, ridge feature-based, correlation-based, and gradient-based. Even though, most of the fingerprint recognition methods are minutia-based, the noise and distortion occurred

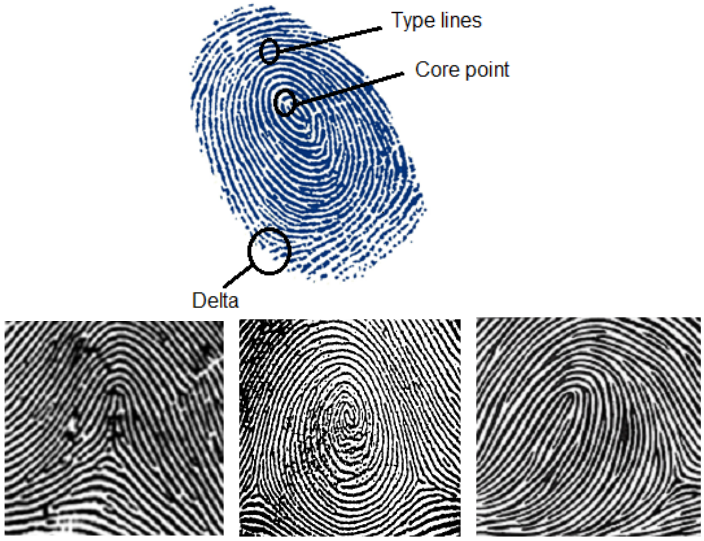


Figure 2.6: **Top:** Fingerprint minutia. **Bottom:** Arch, whorl and loop fingerprint features.

during capturing the image via the sensor results in some missing number of minutia details [80]. Thus, ridge feature-based techniques are used to overcome this limitation [32]. Ridge feature-based methods generally suffer from low discriminative capability, hence, correlation-based methods are employed to do texture correlation analysis between pixels on different alignments of two superimposed fingerprint images [134]. All of these techniques are sensitive to skin condition, finger alignment and amount of pressure, so the gradient based approach which does not require the minutiae for alignment was proposed by Aggarwal et al. in [9]. The authors used a technique based on subdividing each minutiae into many local regions which are being used afterwards to compute the histograms of oriented gradients (HoGs) for characterizing textural information for each minutiae location.

Coetzee and Botha [37] were the first to work on fingerprint recognition operating on images of low quality. They suggested to obtain a binarized image by using both the edge extracted image along with the gray-scale image obtained by a recursive approach tracing lines and their thinnings. The window was first centered on the pixel with the lowest gray-scale value, and then changed its position to trace boundaries of the ridge. The authors of [154] improved feature extraction from low quality fingerprint images by adding noise to the original images. This phenomenon

is called stochastic resonance where the added noise to the original signal increases the amplitude to noise ratio.

One of the most powerful filters in noise removal is the Gabor filter which became the most widely used approach in fingerprint recognition. It was first proposed by Ratha et al. in [147], where a Gabor filter was used to create smooth ridges from the noisy images of the fingerprints, and then a binary image was created using thresholding. This approach was still considered insufficient for low quality images as it could not detect local directions or ridge flow distances.

Various leading studies on fingerprint recognition using neural networks were conducted, and the work by [194] is one of the very leading researches in the area. Two neural networks were used; one for feature extraction and the other for classification. This method achieved 95% accuracy rate.

Furthermore, deep learning also got traction in fingerprint recognition systems. Yao et at. [204] proposed a method that combine two machine learning approaches; Support Vector Machine (SVM) and Recursive Neural Network (RNN). Researchers have studied the combination of both the flat and the structured representation for fingerprint recognition and tested their new mixed approach on the NIST-4, a fingerprint database, obtaining a 95.6% accuracy rate Therefore, the results indicate the high efficiency of using a combination of two different representations.

2.1.3 Iris Recognition

Iris recognition is concerned with the biometric details of the ring shaped area surrounding the pupil of the eye. It consists of two layers; the central pupillary region, and the outer ciliary region, separated by the Collarette. Iris has been proven to be unique in each person and even from the left to right eye of the same person [25].

Generally, Iris recognition systems have a lower False Acceptance Rate (FAR) compared to other biometric modalities, but still have a high False Reject Rate (FRR).

Bowyer et al. [25] summarized iris recognition approaches in three main categories; the Flom and Safir patent [52], Daugman's approach [41], and Wildes's

approach [193]. Flom and Safir suggested in their unimplemented iris recognition system patent, the use of illumination to increase and decrease the size of the pupil to reach a predefined size. The authors recommended the use of an algorithm to find large groups of connected pixels with intensity amounts less than a threshold. Daugman mentioned in his research [41] that in order for the illumination to be controlled and the possess been unintrusive to people, the use of near infrared illumination is a must. Daugman's system relies on positioning the eye in front of the camera, and either maximizing the spectral power in the upper and middle frequency bands of the 2D Fourier spectrum by amending the focus of the system, or giving an audio command to adjust the position i.e., tilting the head or moving it to another position.

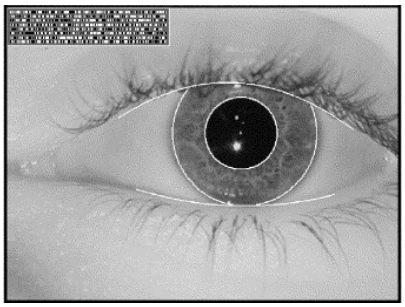


Figure 2.7: Example of an iris pattern, imaged monochromatically captured at a distance of about 35 cm [41].

Daugman tested his algorithm with the United Arab Emirates's border agency using their database of 632,500 different iris images, to check visitors denied access to the country. The False match was found to be Zero [43]. Meanwhile, Wildes used for his approach a low light camera with a wide source and polarization in [193]. The author calculates a binary map and then finds the circles of the iris using the Hough transform, then a Laplacian of Gaussian filter is applied to compute a normalized correlation as a similarity measure.

2.1.4 Other Types of Biometric Recognition

Finger-Vien Recognition

Some human biometrics may change over times due to age or any other circumstances. Palm vein, dorsal hand vein, and finger vein are common vein features which can be used for personal recognition as these are considered as lasting unique biometric details of people. Duo to the chemical properties on the blood, infra-red light can be absorbed in the vein more than by tissues around it, thus the structure of the vein can be captured either by infra-red image sensor or by the reflectance of infer-red illuminations.

Since vein is an inner part of the body, it cannot be fabricated; it cannot be easily recorded or recaptured, also it cannot be artificially re-made from any other material. For these reasons, vein recognition especially finger-vein recognition was given considerable attention in the last few years [190].

Figure 2.8: The exact place of hand vein extracting [48].

There are two main vein recognition methods; structure-based and feature-based methods. Structure-based method depends on the structural characteristics that can be extracted from vein patterns like points, curves, lines, or the structure of the vein net [75,126,187,207]. In feature-based methods, vein patterns are considered as textures and feature extraction algorithms are used for vein feature representations [27, 65, 103, 203].

Gait Recognition

In recent years, different techniques have been proposed for human identification by gait recognition, that is, by studying walking actions of humans and recognizing individuals by variations in their motion patterns. Gait recognition techniques can

be either model-based or model-free. Little and Boyd [110] and Lee et al. [105] both used the shape of the motion to recognize people by their gait. Sundaresan et al. [174], Huang et al. [78], and Sarkar et al. [155] worked on model-based approaches for recognizing gaits. Man et al. [121] proposed a model-free new spatio-temporal gait representation called Gait Energy Image (GEI).

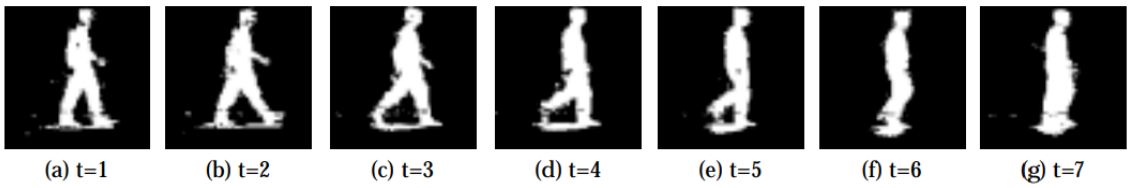


Figure 2.9: A sample sequence of the silhouette of a walking subject after background subtraction [105].

2.2 Multimodal Biometrics

Multimodal Biometrics use more than one biometric trait for single detection purposes. The more identification systems combined together, the more secure and more accurate the results are considered. Various multibiometric approaches have been presented in the last couple of decades [73, 167]. Ross and Jain [151] were pioneers in using multimodal of biometrics for the purpose of identity recognition. Many other hybrid approaches have been implemented; fingerprint and iris features by Besbes et al. [22], multimodals of fingerprint, face and finger veins by He et al. [71]. Moreover, the authors in [55] studied a combination of three different features for their BioID identification system; face, video and lip movement. BioID has been tested on 150 persons for a three month duration and found to be reducing the false acceptance rate to below 1%, depending on the security level. The higher the security level, the higher the false-rejection rate.

Choudhury et al. [35] evaluated a combined system of both face and speaker identification using a Bayesian network. For the face recognition, they combined face detection, head tracking, and eigenface recognition, thus it is achieving a high recognition accuracy. The reliability of each method separately could be predicted

by the derived confidence score. Yang and Ma [200] implemented matching score fusion focusing on three multimodals for identity verification; fingerprint, palm print, and hand geometry. The authors of [83] presented a recognition method using both fingerprint and finger vein for enhancing the score level fusion.

2.3 Attacks on Biometric Systems

Both physiological (face, fingerprint, iris, etc.) and behavioural (speech, handwriting, gestures, etc.) biometric traits are becoming more popular than the traditional methods where pin codes, pass keys, etc. are required to gain access to systems [10, 24]. As biometric based security applications are becoming increasingly popular, the study of their vulnerabilities and the development of countermeasures has become a research topic of current interest. Obtaining illegitimate access to a biometric systems falls in one of two categories; *direct* or *indirect* Attacks. *Direct attacks* are based on theft of biometric data and are carried out against the sensor using synthetic traits such as stolen biometric data of some form; digital images displayed on a screen, printed face or iris images, or in physical form, for example a gummy fingerprint or a fingerprint on a gelatinous membrane. In contrast, *indirect attacks* are carried out against some of the algorithmic modules of the system to construct an input biometric information, as for example through the use of Hill-climbing algorithm and Bayesian statistics [59].

In [148], eight possible attack-points against biometric verification systems have been identified. They are illustrated in Figure 2.10. Each point of attack is vulnerable to a certain kind of spoofing technique. Here, following that paper we are grouping the proposed attack points into two main groups; *direct* and *indirect*.

- **Direct attacks:** where synthetic biometric samples are generated. These samples can be for example; images of a face, fingerprint, or iris. Direct attacks occur at sensor level, level (1) in Figure 2.10, and do not require any pre-knowledge by the attacker of the inner parts of the verification system, such as the used matching algorithms, the feature extraction methods, etc. Besides, these attacks are carried outside the digital limits of the system and

thus digital protection techniques such as digital signature or watermarking cannot be used.

- **Indirect attacks:** These attacks require some extra information about the working of the recognition system. Indirect attacks work at the inner modules of the system in three different attacking modes: (i) changing the information in the communication channel between parts of the system, (ii) attacking either the feature extraction of the matching algorithms using Trojans to bypass certain modules of the system, and (iii) attacking the database of the system and manipulating the data in a form to gain suitable access to the system. Indirect attacks occur at the remaining seven points of attacks in Figure 2.10.

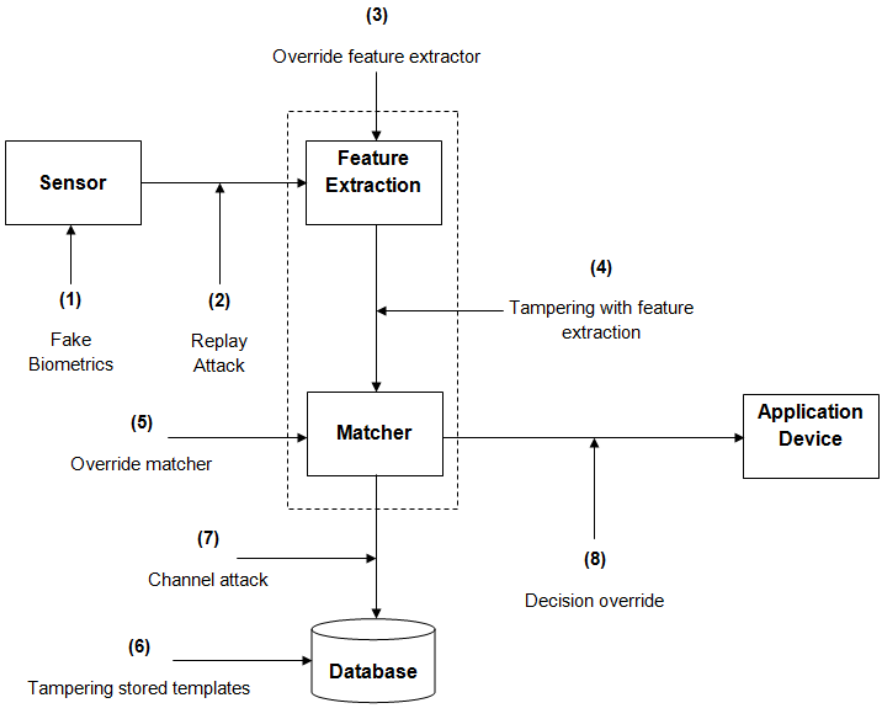


Figure 2.10: The eight possible attack points on the architecture of an automated biometric verification system.

In a more detailed description of the eight types of attacks shown in Figure 2.10:

1. Fake biometrics at the sensor level; where a biometric reproduction is presented in front of the sensor to gain access to the system.

2. An old stored digital biometric signal being resubmitted to the system bypassing the sensor, e.g. an old copy of a biometric image or a previously recorded audio or video. This attack is commonly given the name "Replay attack".

3. Overriding feature extraction; e.g. this could be on attack by a Trojan horse producing feature sets determined by the hacker.

4. Tampering with the feature extraction; this means replacing the extracted features with a different reformed input signal to produce a different synthesized feature set when the representation is known. This attaching mode is considered somehow difficult, due to the inseparably between the feature extraction and matcher stages of the biometric recognition system. Nevertheless, the threat becomes very real when the extracted feature is transmitted to a remote matcher, which could be done over the Internet.

5. Overriding matcher stage; attacking the matcher to directly produce an artificial incorrect match score of lower or higher value.

6. Tampering with stored templates; the attacker amends templates in the database which can be either locally or remotely distributed over several servers. This might result in either unauthorised access of a fraudulent person, or service denial for the correct individual.

7. Channel attack between the stored template and matcher stages; this channel might be attacked and the content being changed on its way back from the stored template to the matcher.

8. Decision override; the final decision of Yes/No is being changed with the choice of the attacker accordingly. This is considered as the most dangerous attack as it most directly affects the final outcome.

2.3.1 Attacks on Face Recognition Systems

Spoofing occurs on various attack-points of the architecture of any biometric verification system. Most major direct attacks happen at the sensor level, the first point

of the architecture. The most common attacking methods are based on the ability of generating synthetic face biometric samples, and spoofing the falsifying system using, for example photos, videos, and/or 3D models [198], to gain illegal access. See point (1) in Figure 2.10.

Pan et al. [139] classified direct attacks to face recognition systems into three categories: a photograph of the real user is used; a video; or a 3D model. One particular strength of such direct attacks is that they do not require any knowledge of the face recognition system they attack. We review these kinds of attacks in detail and give some experimental results on their efficiency against some popular commercial level systems in Chapter 4.

Hill Climbing Attack

The Hill-climbing attack is an automated intrusion technique to generate a positive classified input by analysing match score data returned from false input data. The Hill-climbing attack can be classified within the second or the fourth class of possible attacks on the architecture of the biometric system depending on the point of the attack [123, 148, 182]. In other words, hill-climbing aims at reaching the verification threshold by using the matchers returned similarity scores to modify the template(s) used for the attack.

A hill climbing attack to a face recognition system was proposed in [6] where authors reconstructed the image of a user with the help of face recognition match score results. This technique showcases the fact that allowing access to biometric matching results can implicitly but successfully allow access to the source images. In Adler's research paper [6], the input image face is amended using an independent set of Eigenfaces until the intended match score is obtained. The success rate of the attack is not stated.

Bayesian Hill-climbing Attack

An indirect attack based on the hill-climbing algorithm and Bayesian statistics was proposed in [56, 57]. Figure 2.11 shows an example of successful attack using a hill-climbing algorithm starting from a random face. Iteratively, these faces are modified

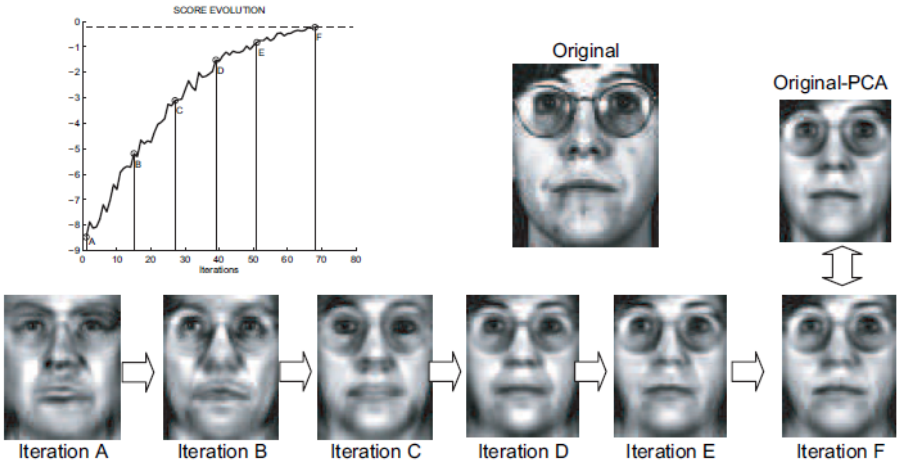


Figure 2.11: An example of the evolution of the score and the synthetic eigenfaces through the various iterations of the attack [57].

to be as close as possible to the PCA projection of the original targeted face. It is noticeable in Figure 2.11 that the final image is visually quite similar to the original face.

Galbally et al. [59] tested the robustness of two verification systems; the eigenface based verification system and the advanced Gaussian Mixture Model (GMM), against the Bayesian hill climbing attack algorithm. Their attacking technique modifies a global distribution computed on the set of all users, by learning the local specificities of the client being attacked. They report success rates of over 85% in by-passing the system for all attacks, even without the use of the real images to initialize the algorithm. The experiment was carried on the XM2VTS database.

While such indirect attacks are perhaps more interesting than direct attacks from a theoretical point of view, however their applicability is limited by their unrealistic assumption that the system is leaking information to the attacker in the form of, for example, face matching scores.

2.3.2 Attacks on Other Biometric Recognition Systems

Attacks on Fingerprint Recognition Systems

Fingerprint recognition systems are vulnerable to spoofing attacks using artificial fingers made from different materials; play-doh, silicon, gelatin, moldable plastic,

or latent fingerprints [5, 160, 169]. In [51, 148, 205], the authors refer to *Fingerprint obfuscation* which is a common term for the alteration of fingerprints by cutting, burning, abrading, or removing part of the skin of a fingertip. *Impersonation*, the creation of a new identity, studied by Valenica and Horn in [183], can be considered as another fingerprint spoofing method.

Marasco and Ross classified fingerprint spoofing methods into two groups, depending on the role of participants; cooperative and non-cooperative methods. A method is considered cooperative if the real client participated in the process of creating the spoof fingerprint. In contrast, the noncooperative method does not require any collaboration from individuals and the process can still run without the presence of the real client. Next, we are giving brief examples of both cooperative and noncooperative methods [122]. See Figure 2.12.

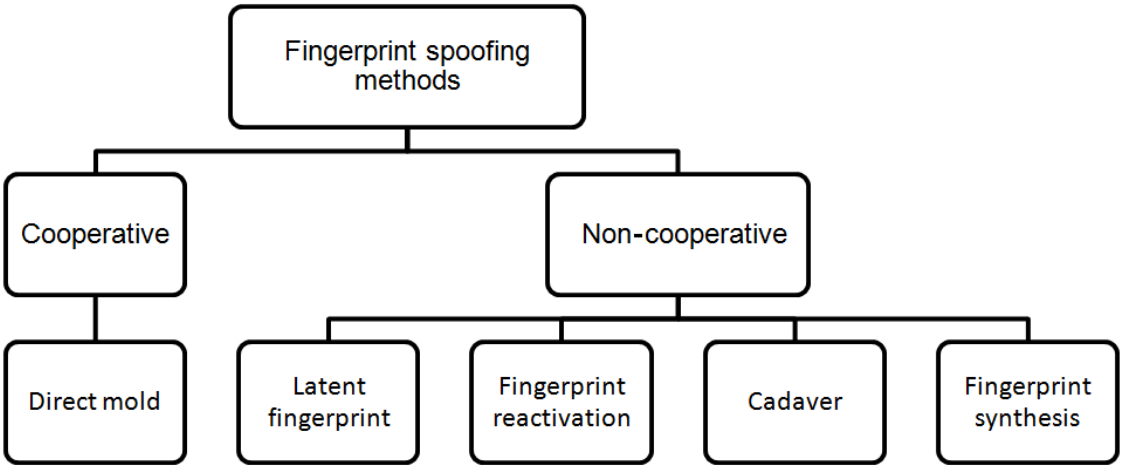


Figure 2.12: A hierarchy of fingerprint spoofing methods.

- **Cooperative duplication**
  - Direct Mold: the artificial fingerprint is created from live finger mold, where the real finger is being pressed on a mold surface, i.e., plaster, dental impression material, latex, or gelatin. The mold or the resulted negative of the produced impression is then filled with some moisture-based material to produce the spoof, e.g. it can be gelatin or any other liquid silicon material.

- **Noncooperative duplication**
  - Latent fingerprints: There are many ways for revealing latent fingerprints. One method is based on latent fingerprints left on any transparent surface lifted with a powder when the real client -for any reason- touched that surface. Latents can be removed from the background carefully using a Scotch tape. This tape which now has the spoofed fingerprint on it is placed on the sensor. Another method, the latents are also lifted from a black powder placed on a transparent surface, then photographed using a digital camera and printed on a transparent object creating a mask which is then placed on a Photolithographic printed Circuit Board (PCB) and exposed to UV light. The resulted plaster is filled with liquid silicon rubber to create a very thin gummy spoof fingerprint which is then attached to a live fingertip and placed on the sensor.
  - Fingerprint reactivation: trying to retrieve the latent fingerprint left by the real client on the sensor by usually simple means i.e., breathing, placing a plastic bag filled with water on to of it, or brushing some powder on the sensor.
  - Cadaver: using a dead finger.
  - Fingerprint synthesis: refers to reversibility of minutiae templates, which means that a fingerprint image can be reconstructed from the template of the enrolled user in the system. This minutiae-based fingerprint can be then transferred to a manufactured artifact.

Putte and Keuning [184] created several sets of dummy fingerprints with and without the cooperation of the real client and used these sets to test various fingerprint sensors against spoofing. This experiment, which took only few hours, found that attacking the sensor level of the fingerprint recognition system with fake fingerprints produced with the cooperation of real clients was more efficient than using fingerprints generated without the cooperation of real clients. Furthermore, a successful attack was reported by Matsumoto et al. [124] against 11 different fingerprint recognition systems, using artificial gummy fingerprints of cooperative users. With

a plastic mold and a gelatin leaf, the authors of [124] spoofed 11 verification systems with a probability in the range 68-100%. Meanwhile, for the same experiment, this percentage falls down to only 67% with non-cooperative clients. Figure 2.13 shows samples of real and fake fingerprints and the average number of acceptance for each device out of the 11 tested verification systems.

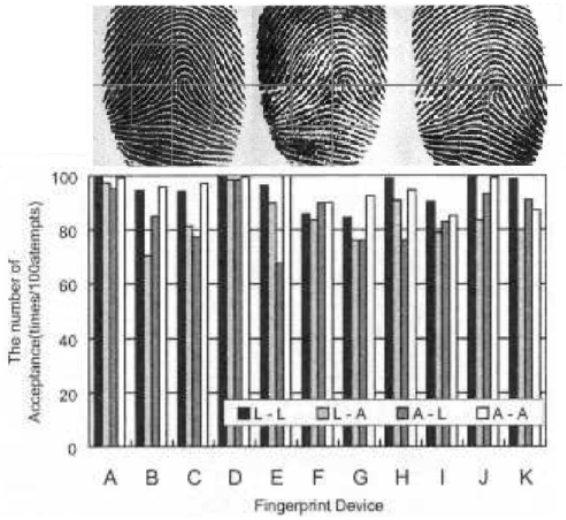


Figure 2.13: Row 1: Example of live, silicon, and gummy fingerprints, respectively. Row 2: The average number of acceptance for each of the 11 tested verification systems based on 5 tested subjects [124].

Martinez-Diaz et al. [123] have tested the performance of the Hill-climbing attack on the NFIS2 system and on a Match-on-Card embedded system. NFIS2 is a reference PC-based fingerprint recognition system, whereas the MOC is the hardware based system. The technique proved the robustness of the NFIS2 against the Hill-climbing attack, but on the other hand, it required high number of iterations, more than 5000. Although Brute-Force attacks were more efficient with the NFIS2 than Hill-climbing attacks, the latter still requires less resources than the former.

Attacks on Iris Recognition Systems

Iris recognition is considered as one of the most reliable biometric modalities, even though attacks to gain unauthorised access to such systems are still possible. These are being done using several techniques such as using printed images, wearing patterned eye contact lenses, using artificial glass/plastic eyes, or displaying video sequences on a tablet or mobile screen in front of the sensor. Several countermeasures

have been proposed by various researchers [39, 40, 42].

Raghavendra and Busch [145] explored the vulnerability of various baseline iris recognition systems against malicious attacks. These tests were carried out using their relatively new iris artefacts database (VSIA database) which comprises of 550 real and 2750 fake iris sample images.

Ruiz-Albacete et al. [152] created real and fake iris datasets containing images from both eyes of 27 users in the BioSec dataset. They had two sessions, each creating four images from each user, which summed up to a total of 432 fake images. And then it was used to test possible attacks at the sensor level of iris recognition systems. The authors experimented with a publicly available iris recognition system in two attacking scenarios; the first is to enroll and later to access the recognition system using the fake image dataset, and the second scenario is to enroll the system with original iris images and then try to gain access with fake images. Both scenarios showed high vulnerability of the recognition system. The experiment resulted in a 40% success rate in spoofing the iris verification system with fake images.

As well as direct attacks, researchers addressed the problem of indirect attacks. Although indirect attacks on iris recognition are rare compared to other biometrics due to the complex pattern of the iris, some research in the area nevertheless exists.

Gomez-Barrero et al. presented in [63] the first evaluation of indirect attacks on a multimodal system based on face and iris recognition. They proposed two algorithms: (i) the hill-climbing attack based on the uphill simplex algorithm to attack real-valued matching scores, and (ii) hill-climbing attack based on a genetic algorithm to use against binary matching scores. Their evaluation showed a remarkable effectiveness of the attacks, proving the vulnerability of a multimodal recognition system based on both the face and the iris, under spoofing and highlighting the fact that even multimodal systems are vulnerable to indirect attacks as well as single modality systems.

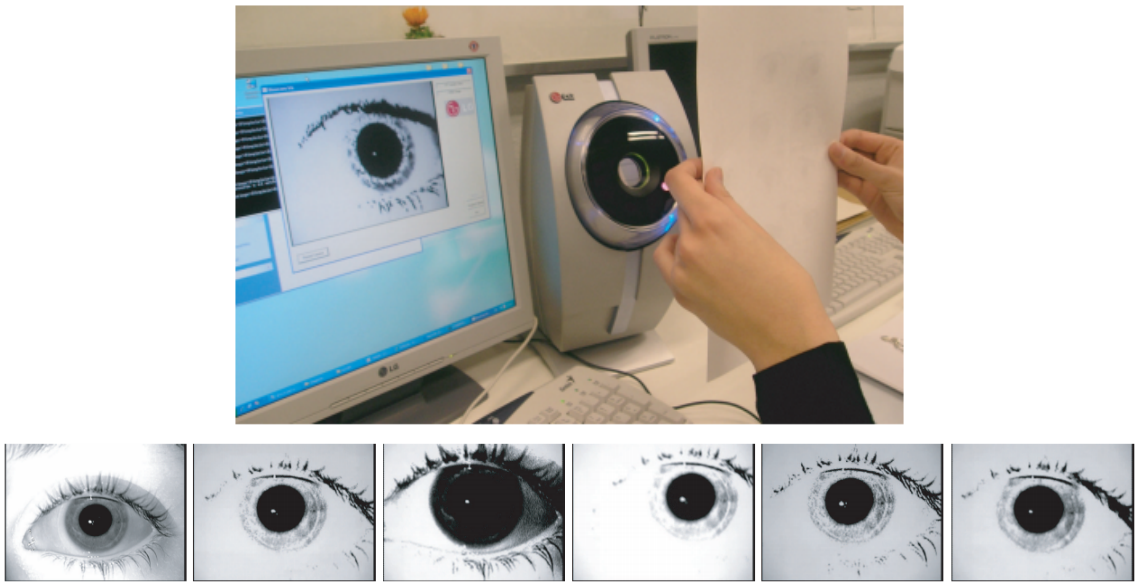


Figure 2.14: **Top**: The process of capturing fake iris images. **Bottom**: captured fake images with various image processing modification printed on high quality paper on an inkjet printer; **left to right:** Original image without enhancement, fake image without enhancement, fake image with histogram equalization, fake image with noise filtering, fake image with TopHat transform, and fake image with Open+TopHat transform [152].

Chapter 3

Liveness Detection

In the last couple of decades, spoofing attempts on biometric systems became a serious problem. Thus, researchers in the area have developed various techniques to counteract these attempts and eliminate the risk of unauthorised access to biometric verification systems. In this chapter, we broadly review the most robust anti-spoofing techniques for various biometric traits. Then, we provide a detailed survey of the relevant face liveness detection methods and finally we discuss in detail the currently commonly used facial spoofing databases. A more detailed review of the [177] paper by Tan et al., which played a key role in informing our investigation, and a discussion of their implementation approach is also included. Finally, we provide more detailed description of the fundamental classification techniques used in this work; Sparse logistic regression and Neural networks.

3.1 Liveness Detection

3.1.1 Face Liveness Detection

Liveness tests are binary classification algorithms distinguishing between live faces in front of the camera and imposter images, videos or 3D models. They have been developed as countermeasures spoofing attacks at the sensor level of face recognition system. In the literature, the term *liveness detection* is used almost interchangeably with the term *anti-spoofing* in the context of attacks at the sensor level of a biometric

verification system. Even though sometimes the term liveness test is not entirely appropriate, e.g. an attack on a face recognition system using makeup, or an attack on an iris recognition system using contact lenses.

In the last few years, liveness detection techniques have been developed rapidly, although it is clear that these cannot be considered a mature technology yet. Indeed, using imposter still images, which is the easiest and most basic way to attack a face recognition system, are still very effective against consumer level of the face recognition system. Figure 3.1 illustrates classification of possible attacks on a face recognition system.

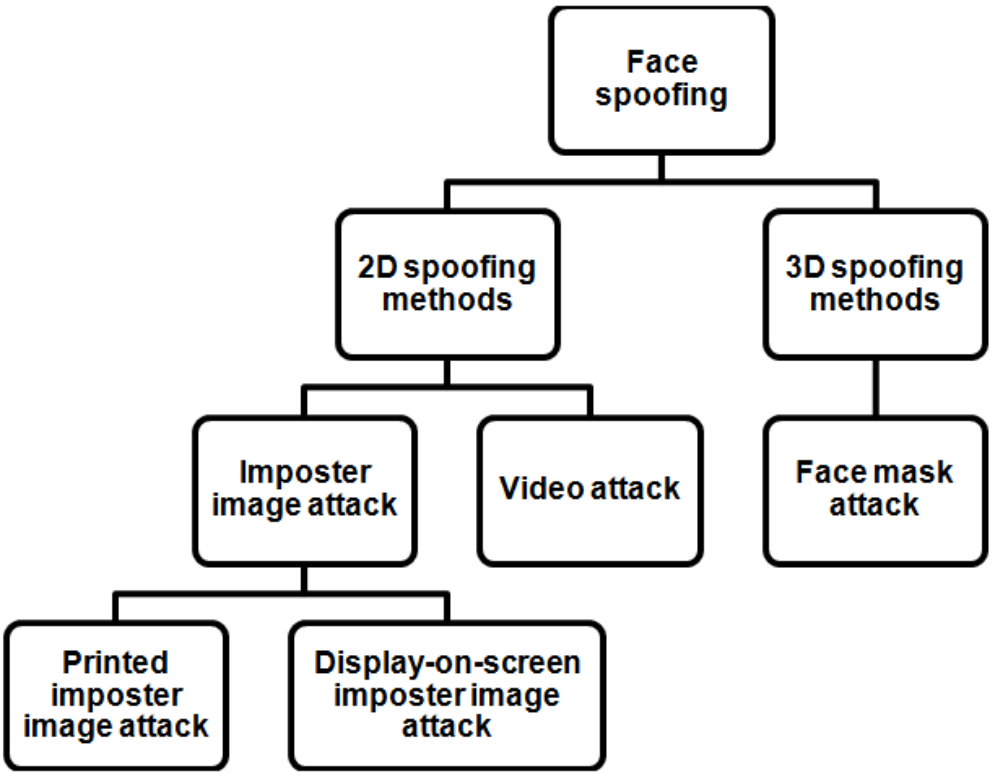


Figure 3.1: Various face spoofing attacks.

Liveness tests use the sensor input of a face recognition system, usually com

mon video, and perhaps some other forms of input such as images from an infrared camera, and aim at distinguishing between those input objects that are coming from alive faces and those that are not. In our research, we classify the current liveness

detection approaches into five main groups: Frequency and texture based, camera focus variations based, motion based, 3D structure based, and liveness detection using special hardware.

Frequency and Texture Based Face Liveness Detection

Working with printed face images, Määttа et al. [116] applied a multi-scale local binary patterns (LBP) to analyse texture features which were being used to detect imperfections in the quality of printed faces. Määttа et al. combined three LBP configurations ($LBP^{u2}_{8,2}$, $LBP^{u2}_{16,2}$, $LBP^{u2}_{8,1}$), after which the resulting multi-scale LBP feature vector is fed into a nonlinear SVM classifier to distinguish between client and imposter images. The proposed technique resulted in (2.9% EER) on the NUAA dataset [177]. Määttа et al. approach has been found robust, computationally fast, and does not require user-cooperation, in contrast to other previous work in the area. Figure 3.2 illustrates Määttа et al. proposed approach where face images are being detected, cropped and normalized to 64×64. Next, $LBP^{u2}_{8,1}$, $LBP^{u2}_{8,2}$, and $LBP^{u2}_{16,2}$ are applied and the result is divided on 3×3 neighbourhoods regions. Then, the three histograms are concatenated into one and finally, an SVM classifier is being used for determining liveness of the face.

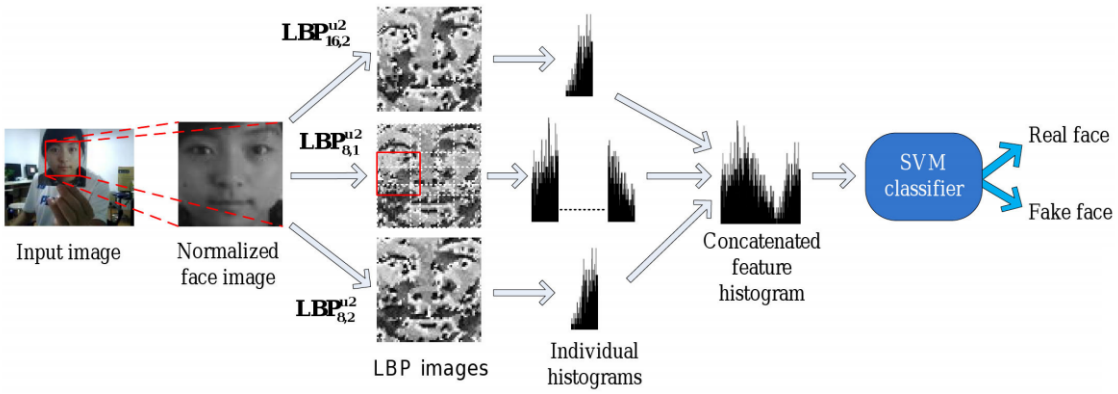


Figure 3.2: Illustration of the Määttа et al. approach [116].

Another work by Määttа et al. [117] used three enhanced feature histograms to encode texture and the gradient structure of face images. A kernel map was then applied to transform the feature vector into a compact linear representation, which

was then used with a fast linear SVM classifier for a successful image classification outcome.

Kim et al. [89] proposed an approach based on differences in shape and detailedness to differentiate between real (3-D) and imposter (2-D) faces. The authors of the paper used a method based on frequency and texture analysis to distinguish live faces from imposters. They used the power spectrum for frequency analysis, exploiting information residing in both high-frequency and low-frequency regions. Two reasons were given for using the frequency analysis; first, the fact that the generated illumination components of 3-D shape images are different in the low-frequency regions. Second, the difference in the detailed information between real and imposter faces leads to variation in the high-frequency information. 2-D images suffer from the lack of texture information compared to images taken from 3-D shapes. In their approach, three feature extraction methods are being implemented; fequency-based, texture-based, and fusion-based feature extraction. The Discrete Fourier Transform (DFT) was used to transform facial images into the frequency domain for frequency-based feature extraction. The frequencies form several groups of concentric rings, each representing the equivalent region in the frequency category. Then, a 1-D vector of features is formed, that combines the average energy values of all concentric rings. LBP is used to extract texture information from the images for texture-based feature extraction. Finally, a Support Vector Machine classifier utilizes the outputs of both the Discrete Fourier Transform and the LBP methods and produces decision values. The authors of [89] have used two databases to evaluate their method; the BERC webcam database and the BERC ATM database. The fusion-based method resulted in a 4.42% error rate compared to the 5.43% frequency based, while the LBP-based method had a 12.46% error rate.

Li et al. [106] proposed a liveness test based on the different distributions in the frequency domain of light reflected from a flat 2D surface, against a 3D surface. Their method is based on the analysis of the Fourier spectra of either a single or a sequence of face images. It mainly focus on the structure and movement information of a real face. Their algorithm is based on two assumptions: the size of a good quality printed imposter image is usually smaller than that of a real face, and the

high-frequency components of the fake photos are smaller than those of the real face images. Moreover, the standard deviation of the frequency components in a sequence of images is small even if the camera is in motion as the poses and expressions of the faces do not vary.

Chingovska and her co-authors [33] introduced their own face spoofing attack dataset (the REPLAY-ATTACK) which contains printed photographs, and photos and videos displayed on digital screens of different sizes. They implemented Määttas work and tested it on three different datasets; NUAA, REPLAY-ATTACK, and CASIA. The HTER was 15.16%, 19.03% and 18.17%, respectively.

Yang et al. [202] proposed a method for liveness detection based on four steps: (1) segmenting the face into various components (mouth, eyes, etc.); (2) coding low-level features such as LBP, HoG, and Local Phase Quantization (LPQ) for each component; (3) deriving the high-level face representation by grouping codes with weights obtained from Fisher criterion; (4) feeding the histograms from all components into a classifier for identification. In their method, real faces are captured once, while imposters are recaptured whether from a photo or from digital display. The authors of [202] noted three main differences between client and imposter images: (1) imposters look more blurry than clients since recaptured images are of limited details compared to captured real objects; (2) there is a variation in face appearances caused by a reflectance change due to the Gamma correction of the camera; (3) abnormal shading on the surface of recaptured images. The authors first detect the whole face (H-Face), which is later divided into six parts: facial, contour, right eye, left eye, nose, and mouth regions. Facial and contour regions are then divided into 2×2 grids to end up with 12 components. Dense Low Frequency local features are extracted from these components to perform component-based on coding to obtain local codes. These codes are then concatenated into a high-level descriptor having weights retrieved from the Fisher criterion analysis. Lastly, these features are being fed into an SVM classifier. Yang et al. [202] tested their approach using three well-known real-imposter facial databases (CASIA, NUAA, and PRINT-ATTACK) and reported better performance on all of the three databases.

The authors in Tan et al. [177], use the Lambertian model to study the dif-

ferences between live human and imposed image relying on two assumptions: 1) live human face is a 3D object, compared to the 2D images of imposters; 2) there is a difference in the roughness of real faces and flat image surfaces. To apply a Lambertian reflection model, the authors proposed two methods for deriving the latent samples: 1) Variational Retinex-based method; 2) Difference of Gaussian-based method. Two extensions to the basic sparse logistic regression model were employed to allow quicker and more accurate classification; 1) Spare low rank bilinear logistic regression, and 2) a nonlinear model based on empirical mapping. For the evaluation of their method, Tan and his colleagues tested their approach on their own, publicly available, large photo-imposter database with more than 50K images from 15 individuals. For more details about their database see Section 3.4 of the thesis. Table 3.1 shows the detection performance of Tan et al. proposed technique.

Table 3.1: Results of Tan et al. [28] approach.

	Min	**Mean**	**Max**	**STD**
Classification Accuracy	85.2%	86.6%	87.5%	0.6%
True Positive Rate	81.9%	82.4%	90.4%	0.6%
False Positive Rate	8.0%	9.3%	18.8%	1.3%

Peixoto et. al [141] further improved the Tan et al. [177] algorithms by addressing the problem of using high-quality printed images and images on digital displays and also considering limitations related to bad illumination conditions. They applied contrast-limited adaptive histogram equalization (CLAHE), which operates on small tiles in the image, before computing differences of Gaussians, increasing the robustness of the method under bad illumination conditions. Then, a standard sparse logistic regression was used, as in [177], to decide whether an image is real or not.

Another anti-spoofing approach, proposed by Komulainen et al. [98], is based on an SVM classification of histograms of gradient descriptors (HoGs). Experiments conducted on two publicly databases, NUAA and CASIA, showed an improved performance over the state-of-the-art.

Galbally et al. [58] suggested a novel software-based liveness detection approach

based on multi-biometric modalities using an Image Quality Assessment (IQA), where lower quality images are by default classified as imposters. Some of expected quality differences between real client and imposter images might be: sharpness level, degree of brightness, color level, local artifacts, and the amount of details found in the image. For instance, a face image captured by a mobile device camera can often be either under or over exposed. The proposed usage of the metric is capable of operating on multi-biometric systems and under diverse spoofing contexts with a very good performance when tested on various databases. It performed better than the state-of-the-art.

Lai and Tai [101] recently proposed a liveness test against attacks by fake images or videos displayed on HD screens by analysing the chrominance characteristics and the saturation of the face recognition system's input images. The implemented mehtod resulted in an outstanding success rate of above than 99%.

Further work by Kim et al. [92] on liveness detection uses the diffusion speed to discriminate between the illumination characteristics of live face and recaptured images. The authors use local pattern of diffusion speed values as linear SVM input features, called local speed pattern (LSP).

In [7] the authors proposed an algorithm that extracts block-wise Haralick texture features from DWT frames obtained from a video. They use PCA to reduce the dimensionality of the feature vector and an SVM is used for the classification problem.

In a very recent work, Chan et al. suggested the use of two images one taken with flash light and another one under ambient illumination. They noted that the use of an additional light source increases the efficiency of texture pattern detection, which enhances the differentiation between real and imposter images. The two input images are analysed using four texture descriptors, LBP of face patches, the standard deviation of the differences of two patches, and the mean and the standard deviation of the differences of the backgrounds [29]. And in another similar recent work by Martino et al. [46], 3D information is extracted from images without full 3D reconstructions, using as additional devices flash light and stereo camera.

Variable Focusing Based Liveness Detection

Kim et al. [91] suggested detecting live faces by studying the differences in two images taken in different (in/out) focus settings. In real faces, focused regions are clear and the rest of the image is blurred, while in printed images, there is little variation in focus due to images being flat. Their method depends on the Depth of Field (DoF) of the camera, and uses a sequence of images to find the extent of focus variation at pixel level. First two sequential photos are being taken with different focus. One is focused on the ears and another on the nose because of the difference in distance from camera lens between nose and ears in real faces. The gap in depth between both sequences is adequate to express the 3-D effect. The Sum Modified Laplacian (SML) is used to compute a value for the degree of focus. The last step is to compute the difference of SMLs, which exhibits similarity in patterns for real faces, compared to fake faces. This difference in patterns between real and fake faces is used as the feature for face liveness detection. The method gave in 2.86% FAR and 0.00% FRR when the DoF of the camera was very small, otherwise, it gave higher average of FAR and FRR.

Motion Based Face Liveness Detection

There are two kinds of motion relevant to face liveness detection; either a set of expressions or movements of the face, or movement of the face in relation to the background. Recent approaches to liveness detection relying on biometric motion analysis, focusing on different types of motion include: head tilting [17], mouth movement [97], holistic face movement [96], and eye-blinking [137]. Foreground and background motion correlation is used in [13].

Eye-blinking is the physiological act of continuous opening and closing the eyelids. A human blinks at least 15-30 times in the minute, with an average of 250 milliseconds for each one blink [88, 179]. Current cameras can capture at least 15 fps, giving frame interval of not more than 70 milliseconds. Therefore, capturing two or more frames for each blink is possible, which allows considering eye-blinking as a clue for anti-spoofing. There are three main advantages of using eye-blinking in anti-spoofing; (1) No extra hardware or special tools are needed, (2) no user col-

laboration is required as eye-blinking is a natural behaviour of human beings, (3) eye-blinking is a distinguishing characteristics between a real face and a facial photo.

Some of the recent work on eye-blinking assumes highly controlled conditions and requires high-quality inputs. Tian et al. [178] use a system of automatic recognition of human facial action units. Moriyama et al. blinking detection method [128] is based on the variation of the average intensity in the eye region, which is sensitive to the illumination conditions and distortion.

Pan and Lao [137] developed an approach in liveness detection by recognising spontaneous eyeblinks. Their eyeblink-based method does not require any additional hardware except a commonly webcamera. The authors formulated blink detection as an interface, where the user interacts with camera through blinking action, and they used an efficient discriminative adaptive boosting algorithm for the purpose of quick and accurate blink behaviour recognition. The use of the undirected Conditional Random Field (CRF) framework developed by the authors to model eye-blinking, addressed the Hidden Markov Model (HMM) limitations. CRF is statistical modelling methods for segmenting and labelling sequence data. Since eye-blinking is an action represented by and image sequences of open and close states of eyes, CRF has been found to be convenient probabilistic method for the purpose of face liveness detection based on the action of eye-blinking.

Sun et al. [170] also proposed a blinking based approach using Conditional Random Fields for modelling blinking activities, using as input an eye image sequence which includes images in both close and open-eyes state. Furthermore, the authors compared AdaBoost and the Hidden Markov Model with the proposed CRFs model. Their approach was tested using a set of live eye-blinking and imposter video sequences, consisting of 80 video clips from 20 individuals, shooting the action of eye-blinking. The first clip of each individual gives a set of frontal face views without eye glasses, the second clip includes a set of frontal face views with thin rim glasses, the third clip includes frontal faces and black glasses, and finally, the last clip contains upward no-glass views. Each clip has a duration of 5 seconds at 30 fps at 320×240 pixels resolution. Another 180 clips of imposters were used, including various motion of the photos, such as rotation, folding, and transition.

The use of an undirected conditional graphical framework to simulate a fusion of eye-blinks clues and scene context clues was proposed by Pan et al. in [138]. The experimental results showed that their system had the ability to counteract common spoofing techniques.

Jee et al. [84] proposed a technique based on the analysis of the eye movement. When an eye is being detected a sequence of input images is analysed to calculate the variation of eye regions. if the result is a shape variation bigger than a threshold, then the input face is recognized as live face, and as an imposter otherwise. For the eye detection, the authors made the assumption that in case that image is a real image then the intensity of the eye regions is lower than the rest of face regions. A Gaussian filter was used for detecting face regions to obtain a smoothed 3D surface. Local minima are then extracted, using gradient descent. Invalid eye candidates are removed by Viola's AdaBoost method, and finally, they use Hamming distance to calculate liveness scores for the eye regions. Their experimental results give a mean score of 30 for live faces and 17 for fake faces. With a threshold set up to 21, method's performance was at 0.01 for FAR and 0.08 as for FRR.

Kollreider et al. [97] proposed the use of lip movement classification, in conjunction with detection based on face landmarks, for face liveness detection. They used an SVM classifier for detecting lip dynamics, located mouth regions and extracted an Optical Flow Lines (OFL) in real time. The authors used the XM2VTS database to evaluate their approach. Each individual in that database was recorded pronouncing the digits from 0 to 9. They used a total of 100 video clips for training and testing; 60 videos for training the SVM classifier and cross validation, and 40 videos for testing. Their method obtained an accuracy rate of 73%.

Singh et al. considered eye and mouth movement for face liveness detection in [162]. They use the Haar classifier to identify both eye and mouth movements, where eye liveness is demonstrated by opening and closing eyes in the considered time interval, while movement of the mouth region is calculated at the mouth's region of interest from detection of the teeth's HSV values.

Kollreider et al. [95] developed a face liveness method using optical flow pattern matching for face part detection, relying on the fact that 3D faces generate a special

2D motion that is more dynamic in the centre of the face parts such as nose compared to outer parts (e.g. ears). Moreover, the observed parts nearer to that camera have different types of motion compared to the parts away from it. In [96], Kollreider et al. proposed a novel approach based on a combination of face part detection and optical flow estimation. They used the Optical Flow of Lines to compute liveness score. OFL is capable of differentiating between motion of points and motion of lines, and was used to model and distinguish between live face movement and still image movement. For face part detection, the authors combined pattern matching with a model-based technique based on Gabor features and an SVM. The classifier evaluation of this approach on a database which contained 100 video sequences from the Head Rotation Shot-subset (DVD002 media) of the XM2VTS database with the data downsized to 300×240 pixel resolution. 200 live and 200 imposter sets were used and their method gave 0.75% error rate.

Chetty and Wagner [31] proposed a multi-modal framework for liveness detection based on the face-voice fusion technique for individual verification. The introduced framework utilises the static and dynamic bi-modal feature fusion, cross-modal fusion, and 3D shape and texture fusion techniques. There, two types of photo attacks were tested; *type-1* replay attacks with still photos and pre-recorded audios, and *type-2* replay attacks of animated videos created from still photos and again pre-recorded audios. The validation of the Chetty and Wanger proposed method resulted in less than 7% EER for *type-1* attacks, while it performs significantly better on *type-2* attacks.

Moreover, Anjos et al. [13] proposed a method based on foreground and background motion correlation. Frischholz and Dieckmann [55] proposed an interactive multimodal biometric system BioID, where the system randomly request different head poses and movements from the user. Similarly Akhtar et al. proposed in [11] an anti-spoofing technique using a multbiometric system.

Bao et al. [17] introduced the use of optical flow field to analyse the characteristics and differences between optical flow fields generated from 3-D objects and 2-D images. These flows are generated from four main movement types: translation at constant distance from the observer, rotation at constant distance about the view

axis, moving backwards and forward, and swing of a planar object to the view axis. The first three movements produce similar optical flow fields while the fourth type creates a significant difference in optical flow fields. Three groups of printed faces were used to conduct the experiment; first, a group of 100 printed face images translated and randomly rotated. Second, a group of 100 images from the previous group folded and curled before being used for testing. Third, a group of 10 individuals's live faces each of them doing ten different gestures like shaking, swinging, etc. With a camera working at 30 fps sampling rate, each captured video had a duration of 10 seconds, and computations were done on images sampled every 10 frames. Their method cannot deal perfectly well with 3-D objects and might have low accuracy when there are illumination changes.

3D-Structure Based Liveness Detection

These types of methods are based on the observation that is based on images, 2D surfaces exhibits a lack of surface shape variation compared to those that come from 3D surfaces. Lagorio et al. [100] suggested an approach relying on the computation of the mean curvature of the surface from the principle components of local Cartesian coordinates. The authors tested their approach on two sets: a fake set and a genuine vectra set. In their experiment the FRR was computed as zero. Another technique on liveness detection by analysing a sparse 3D facial structure was proposed by Wang et al. [189]. It is based on the fact that real images have more 3D structure information, while imposters usually lack in depth. Facial landmarks are detected and key frames are selected to be later used in recovering the sparse 3D facial structure. An SVM classifier is used to differentiate between real and imposter images. To evaluate the technique, three databases were constructed, using different quality cameras, testing the performance of the method across different devices. The measured accuracy of the proposed approach was 100% for the classification and the liveness detection.

Liveness Detection using Special Hardware

Developing accurate liveness tests is a challenging task and very often they require specialized hardware, such as special sensors beyond the visible spectrum. It is well documented that the accuracy rates of liveness tests can be boosted with the use of specialized hardware. For example, consumer level products such as Windows Hello require the detection of an infrared camera in the hardware setup before allowing the user to enable face recognition based user authentication.

There are numerous researches in the area of face recognition anti-spoofing using the Near Infrared Recognition (NIR). Socolinsky et al. [166] and Bebis et al. [19] analysed face thermograms acquired by a thermal imaging camera. Kim et al. [90] proposed the use of raw light field photography to detect spoofing attempts, while Steiner et al. [168] recently proposed a liveness test based on the analysis of the spectral signatures in the Short Wave Infrared (SWIR).

Face Liveness Detection and the Convolutional Neural Networks

Recently, Convolutional Neural Networks (CNNs) have been used to learn features for face anti-spoofing, instead of using hand-crafted features such as LBP, HOG, LBP-TOP, or DoG. Yang et al. [201] used a CNN to learn features of high discriminative ability in a supervised manner, instead of using manually pre-designed feature extractors. Their results show a decrease of HTER by 5% on two widely used datasets, CASIA and the REPLAY-ATTACK, compared to the state-of-art.

Li et al. [107] proposed a novel approach based on using the deep part Convolutional Neural Network. The authors firstly fine-tune the CNN on the face spoofing database, and then to avoid the overfitting problems, the PCA method was utilized to reduce the dimensionality of the feature space. Finally the SVM was used to differentiate between real and imposter faces. Two spoofing databases were used in validating this approach, CASIA and the REPLAY-ATTACK. The approach achieved a better performance than the state-of-the-art.

Alotaibi and Mahmoud [12] also used a CNN for liveness detection. They applied an Additive Operator Splitting (AOS) based scheme to detect spoofing attacks. Using a large time step they extracted the sharp edges and texture features, respecting

of the boundaries location of the input image. Imposter images have less sharp edges and more flat surfaces around the eyes, nose, cheek, and the lips regions, compared to the real images. This is particularly true around the nose and lips. Alotaibi's and Mahmoud's approach achieved a 10% HTER with a time step of $t = 100$.

This promising research direction seems at the moment to be hampered by a lack of face anti-spoofing specific data for CNN training. Thus, for our research, we either opted for a simpler ANN based generic algorithm which was sufficient for some purposes, or used a pretrained CNN.

Despite the large number or researches in the area of liveness detection for face recognition systems, no research was found that considers spoofing attempts using processed imposters in any form. All researches were considering plain recaptured images or videos from real faces, without being altered in any way. Hence, in our research we decided to draw attention to the importance of considering attacks made with processed images or videos.

3.1.2 Anti-spoofing Methods for Other Biometric Recognition systems

Fingerprint Anti-spoofing Detection

Fingerprint spoofing has been found to be possible and relatively easy compared to other biometric modalities such as iris biometrics. Hence, many researches were conducted for developing anti-spoofing mechanisms to counteract spoofing attempts. Some of these anti-spoofing mechanisms rely on using an additional hardware devices for retrieving information such as: temperature, heartbeat, blood pulse, pulse oximetry, odors, or detecting a real finger surface by scanning of laser means, or based on the electrical conductivity of the skin. Various anti-spoofing methods do not require any additional hardware and depend only on software to obtain information such as: ridge frequencies, textural characteristics of the skin, skin elasticity, perspiration, or the power spectrum of fingerprint image.

Marsco and Ross [122] classified software-based fingerprint anti-spoofing methods into two groups according, to the nature of the extracted features; static and

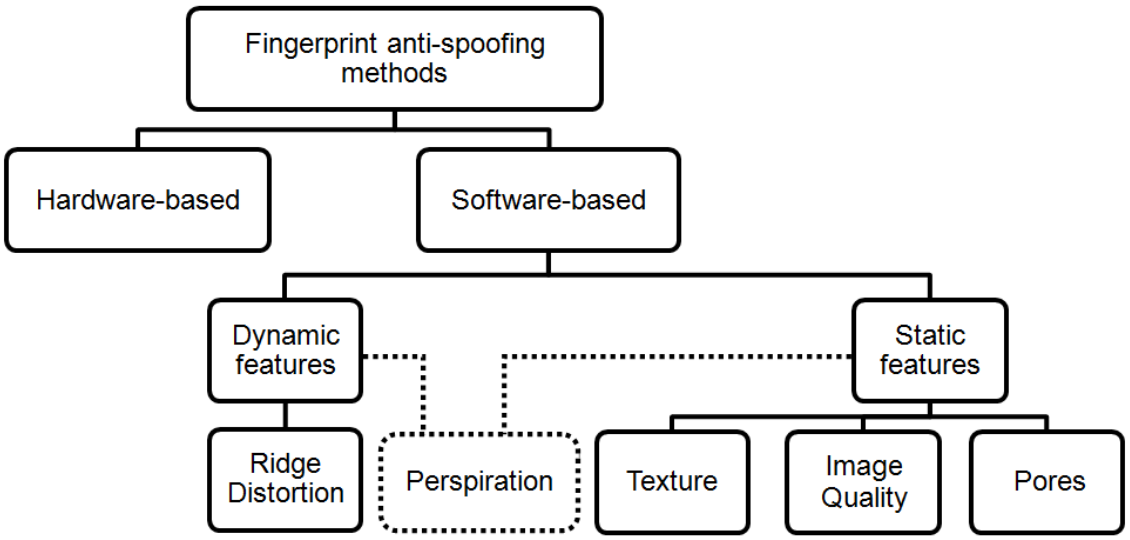


Figure 3.3: Hierarchy of fingerprint anti-spoofing methods.

dynamic. See Figure 3.3.

- Dynamic features. Are those features obtained from several frames of the fingerprint acquisition device.
  - Ridge Distortion Based methods: studying the skin distortion when a finger is pressed against the surface of a scanner. The assumption is that the distortion of live fingerprint ridges is more significant than that of the fake ones. A high frame rate is required to analyze skin distortion by processing a series of frames as a user is applying some pressure on the scanner placing and rotating his/her finger on its surface.
  - Perspiration Based: this technique relies of the observation that areas around pores are getting darker over time, when the sweat starts from the pores and spans over the fingertip, filling the areas between ridges. This moisturising pattern can be captured by observing multiple frames of a fingerprint over a span of time.
- Static features: where only a single image of a fingerprint is required to extract a feature. Such methods are generally cheap and fast.
  - Texture based: several textural properties differ between spoof and live fingerprint images, e.g. morphology, orientation, smoothness, and hence,

textures can be used for liveness detection. Various algorithms can be used to analyse these texture-based features such as: Local Binary Patterns, Local-Ridge Frequency analysis, Power spectrum analysis, Local Phase Quantization, and Weber Local Descriptor.

- Perspiration based: here and unlike the perspiration features extracted several frames, these perspiration based features are extracted from a single frame image. Two such perspiration based features can be extracted; individual pore spacing, and intensity. In individual pore spacing, the gray-level variations caused by the regular periodicity in the patterns of pores on the ridges is studied. In intensity-based features the analysis of the gray captured image scale is usually based on histograms.
- Quality based: a study of the quality of a fingerprint. The measurements can be related to strength, clarity and the continuity of ridges.
- Pore based: pores are detected using two types of filters; high-pass filters which are used to identify active sweat pores and correlation filters which are used to locate the position of pores.

One of the very leading researches in fingerprint liveness detection was done by Derakhshani et al. in [45]. Their line of research uses skin perspiration pattern and study the sweat diffusion patterns and the periodicity of sweat patterns to detect spoofs. Subsequently, Abhyankar and Schuckers [159] applied a wavelet-based algorithm to isolate the perspiration patterns and to extract and analyse multi-resolution coefficients from the real images and imposter.

Fingerprint Liveness Detection with Convolutional Neural Networks:

Most of current liveness detection techniques are based on handcrafted features. These features require a pre-knowledge of the domain of application in order to come out with features that are able to work suitably with liveness detection algorithms. The approach is considered an expensive way of engineering features, and usually not all possible attacks can be handled with it. Thus, many researchers decided to use deep learning to detect spoofing attacks.

Convolutional Neural Networks have been implemented in many domains, and were proved to have great power, especially with local feature extraction on images. Nogueira and Alencar Lotufo [129] were the first to implement liveness detection for fingerprint using the Deep Convolutional Neural Network (DCNN). The authors designed their feature extraction and classification into two separate parts, the first part is a CNN with Local Binary Patterns for feature extraction, while a Support Vector Machine was the final classifier. The validation of this approach was done on a database of 50,000 real and spoof images of fingerprints and achieved an overall rate of 95.2%.

Wang et al. [186] was inspired by the work of Nogueira and Alencar Lotufo. The authors proposed the novel approach in using a DCNN for feature extraction and classification. They divided the image into small patches and used a threshold for identifying background-free training data.

Iris Anti-spoofing Detection

Recently, it was showed that spoofing iris systems is possible by using some relatively simple methods such as printed iris images, or videos, or purpose-made eye contact lenses. For that reason, many researches were done to counteract attempts to iris recognition systems.

Daugman [39] was the first to study attacks on iris recognition systems and used a method based on the Fast Fourier Transform (FFT), where the total high frequency power was computed to assess the quality of fake iris images. The higher the total high frequency power, the most probable the image was genuine, helps identifying the clearer image. Daugman also introduced a novel sheet model of linear stretch to study the changes in pupil dilations. Several other solutions for anti-spoofing protection of iris recognition systems were suggested; some of them rely on the use of special hardware tools [87,102,136] and others are only software-based solution. For example, in [79,94,199], the authors study the use effect of cosmetic contact lenses, while [191] depends on texture analysis for finding the effect of using someone's else iris patterns printed on colour contact lenses.

Among all biometric traits, face liveness detection is the most developed research

area, the reason perhaps being that developing spoofing techniques is relatively easy compared to other biometric systems such as fingerprint or iris recognitions. As a result, liveness detection for other biometric traits rely on the state-of-the-art methods first developed for face anti-spoofing. On the other hand, most of the other biometric traits use special hardware and are deployed in controlled environments, making them more robust against spoofing attacks. Therefore, liveness detection techniques developed for such biometric traits are not easily transferable to the field of face liveness detection.

3.2 Face Spoofing Databases

For evaluating the effectiveness of any implemented liveness detection test, many researchers designed their own facial spoofing databases. Nevertheless, only few databases became publicly available. Here we are reviewing some of these public databases:

NUAA is one of the earliest public-domain face spoofing databases, created in 2010 by Tan et al. [177]. It contains face images from 15 participating people, taken under non-uniform lighting conditions. Participants were from both genders and were asked to assume neutral facial expressions without head movement or eye-blinking. The client image part of the database, consists of 500 coloured images of each participant, at 640×480 resolution, taken by a conventional web-camera with a frame rate of 20 fps. Client images were printed in three different sizes, 6.8cm×10.2cm and 8.9cm×12.7cm on photographic paper and on a 70g A4 paper, using an HP colour printer. Imposter images, were produced from these printouts with a Canon camera from a distance that would allow the face to cover approximately 2/3 of the whole scene. In addition, NUAA only includes images from hand-held camera printed photo attacks.

PRINT-ATTACK is a publicly available database described in [14] containing videos from 50 participants. The videos were captured under two different sets of conditions: controlled and adverse. In the controlled environment the background lighting conditions are uniform, while the adverse environment has uncontrolled

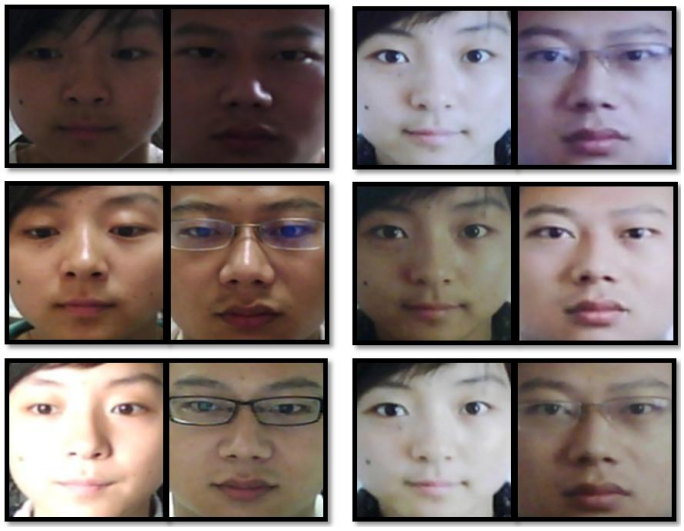


Figure 3.4: Sample of the NUAA database. In each column, top to bottom, the images are from three different sessions under various illumination conditions. In each row, the left pair consists of images of real faces, while the right pair of images of recaptured faces [177].

background lighting conditions. The database consists of a total of 200 real access videos and 200 videos of spoofing attempts, which were made by using A4 printed photos of the participants.

Figure 3.5: Sample of the PRINT-ATTACK database. A printed image of high resolution is used for recapturing [14].

REPLAY-ATTACK the Idiap REPLAY-ATTACK database was released in 2012 by Chingovska et al. [33] as an attempt to enrich and overcome shortcomings of

the PRINT-ATTACK database. It contains 1200 short video recordings from 50 participants; 200 real-access videos were captured by having real clients trying to gain access to the system created with a video acquisition system built on an Apple 13-inch MacBook laptop, and 1000 videos of imposters holding photos or tablets playing video recordings for at least 9 seconds. In total there are two 15-second video clips for each participant of 320×240 resolution at 25 fps. Both real access and imposter videos were taken under the two different illumination conditions. In the adverse environment with a more complex background was used, and the office lights are out, while is the controlled environment the office lights are on, the blinds up, and a homogeneous background is used. For the spoofing attempts, two high resolution shots of each person using a 12.1 mega-pixel Canon PowerShot SX150 IS camera and an iPhone 3GS 3.1 mega pixel camera were taken. Spoofing attempts were classified according to three different scenarios: (i) photos printed on A4 paper (ii) on video and photos playbacks iPhone mobile display (iii) photos and videos displayed on an iPad screen with 1024×768 resolution. The REPLAY-ATTACK database was then used to evaluate the performance of a liveness test based on histograms of LBPs, as proposed in [116], and an SVM as classifier.

Figure 3.6: Sample of the REPLAY-ATTACK database. **Top row:** samples from the controlled scenario. **Bottom row:** samples from the adverse scenario. **Columns from left to right:** real access, printed photos, mobile phone, and tablet photo attacks [33].

CASIA is a database presented by Zhang et al. [208] in 2012. It consists of 600 video recordings of real clients and imposters from 50 different individuals. CASIA

was designed with the maximisation of the variability as its main aim. The client images come in three different imaging qualities: low, normal and high. The spoofing attempts made use of either photos printed on copper paper, or an iPad display. The database consists of 12 videos for each subject, 3 of which are genuine and 9 are spoofs. Even though, CASIA is considered relatively a small database compared to the REPLAY-ATTACK, it contains more diverse samples using a diverse range of commercial devices, i.e., a high resolution Sony NEX-5 camera and a low-quality webcamera, face variations such as pose and expressions, and types of attacking attempts, such as cut photo, wrap photo, and HD video replay.

Figure 3.7: Sample of the CASIA database. **Top left four images:** set (L1_L4) low quality videos. **Bottom left:** set (N1_N4) normal quality videos. **Right set:** set(H1_H4) high quality videos. For each set (from left to right): live face image, wrapped photo attack, cut photo attack and video attack [208].

BERC Webcam Database and BERC ATM Database are two databases, each containing a set of client face images and 4 different sets of imposter face images (photo, print, magazine and caricature). All images in both databases are of the same size 640×480. The webcam database used a web camera to capture the images, while the ATM database used a camera built-in on the ATM [49]. The webcam database's resolution is higher than of the ATM database, since the built-in camera of the ATM has a transparent plastic cover which has an effect in reducing the resolution of the output image. Each database contains images from 25 different subjects. All images are captured under three different illumination conditions: indoors with no additional lights, a strong light towards the front, of the face, and finally a strong light from the side of the face. The Webcam database and ATM

database contain 1650 and 1200 image sequences, respectively. Each sequence image consists of 10 frames.

Figure 3.8: **Top:** sample images from the BERC ATM database. **Bottom:** sample images from the BERC ATM database. The first column shows real face images [89].

MSU MOBILE FACE SPOOF consists of 440 photos and video recordings of genuine and spoofing attempts from 55 individuals using the cameras and displays of a Mac Book Air 13-inch laptop and a Google Nexus 5 Android phone with resolutions of 640×480 and 720×480, respectively. Each video has at least nine seconds duration [192].

The papers describing these databases [14, 33, 177, 208] were the most relevant to the development of our DURHAM FACE database which we will introduce in Chapter 6. We note that all of them, unlike our DF database, are limited to clients and plain imposters without any processing being applied to the recaptured images.

3.3 Binary Classifiers

Machine learning became a popular approach to classification problems, due to its ability to automatically learn and improve from the training data. Machine learning lies at the heart of almost any face liveness detection algorithm.

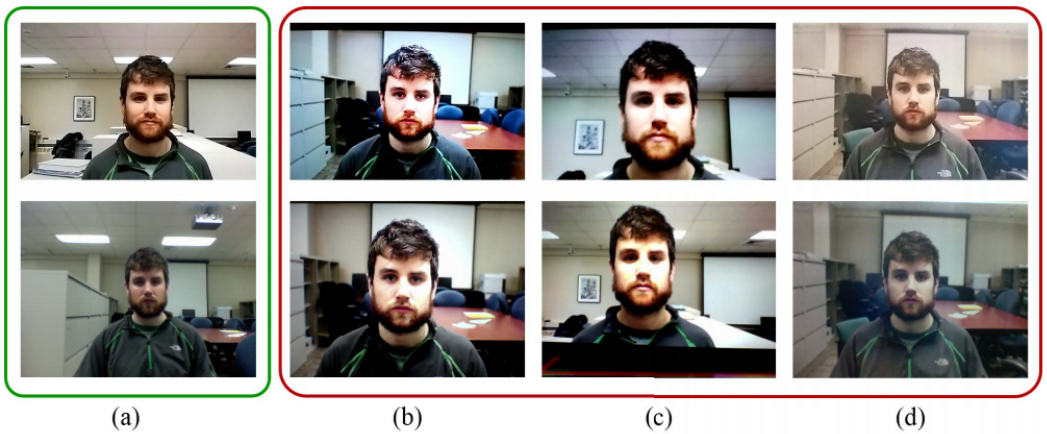


Figure 3.9: Sample images of real and imposter faces from the MSU Mobile Face Spoof database captured using: **Top to bottom:** (i) Google Nexus 5 smart phone camera, (ii) MacBook Air 13-inch laptop camera. **Left to right:** (a) real faces; (b) imposter faces generated by iPad video replay attacks; (c) imposter faces generated by iPhone for video replay attacks; (d) imposter faces generated by printed photo attacks [192].

3.3.1 Sparse Logistic Regression

Logistic regression is a widely used machine learning binary classification technique [93,111]. It has been used on many applications such as bioinformatics [109,144,156, 212,213], computer vision [54], natural language processing [86,119], and document classification [26,30].

Logistic regression is subject to overfitting. Hence, regularization has been applied to reduce overfitting problems and achieve a better generalisation for the classifier. In particular, ℓ_2-Norm regularization has been used on the Logistic regression model in convex optimization problems. ℓ_1-norm regularization has been applied in situations with high dimensional data to obtain sparse models.

Assuming a set of sample data $x = \{x_1, x_2, ..., x_n\} \in \mathbb{R}^n$, and the associated binary class label $a \in \{1, -1\}$, the aim is to predict the label of a new example, using the logistic regression model:

$$\text{Prob}(a|x) = \frac{1}{1 + \exp(-a(b^T x + c)} \tag{3.3.1}$$

where $\text{Prob}(a|x)$ is the conditional probability of x having the label a, $b \in \mathbb{R}^n$ is the weight vector, and $c \in \mathbb{R}$ is the intercept. $b^T x + c = 0$ defines a hyperplane in the feature space on which $\text{Prob}(a|x) = 0.5$. The conditional probability $\text{Prob}(a|x)$ is less than 0.5 if $b^T x + c$ does not have the same sign as a, and greater than 0.5 otherwise. Suppose that we are given a set of m training data $\{x_i, a_i\}_i^m = 1$, where $x_i \in \mathbb{R}^n$ denotes the i-th sample and $a_i \in \{-1, +1\}$ denotes the corresponding class label. The likelihood function associated with these m samples is defined as

$$\prod_{i=1}^{m} \text{Prob}(a_i|x_i)$$

The negative of the log-likelihood function is called the (empirical) logistic loss, and the average logistic loss, defined as:

$$f(b,c) = -\frac{1}{m} \log \prod_{i=1}^{m} \text{Prob}(a_i|x_i) =$$

$$= \frac{1}{m} \sum_{i=1}^{m} \log(1 + \exp(-a_i(b^T x_i + c))), \tag{3.3.2}$$

which is a smooth and convex function. The training algorithm determines b and c by minimizing the average logistic loss: $\min_{b,c} \text{loss}(a, c)$, leading to a smooth convex optimization problem. In sparse logistic regression, we add a $\ell 1$-norm regularization to the loss to avoid overfitting; that is the minimisation problem computes: $\min_{b,c} \text{loss}(a, c) + \lambda \parallel b \parallel$.

3.4 Face Liveness Detection from a Single Image

The way human faces are captured by camera varies duo to different illumination conditions, and the use of cameras with different qualities. Two major differences between real human faces and faces in a captured photos have been identified; (1) real faces are 3D objects while images are 2D, (2) there is a difference in surface roughness between real faces and those in a photo. Using the Lambertian model [133], Tan et al. [177] proposed two methods to identify latent samples by extracting the crucial information about difference in surface roughness between live and recaptured face images; a Variational Retinex-based Method and a Difference of Gaussian (DoG) based method. Live faces taken by a camera are in fact images that have been

captured once, while imposters are photos of a photo, i.e., an image that has been taken twice. Therefore, the distortion in an imposter image is expected to be higher than that of a real human face image, with lower photo quality in the imposter image (less high-frequency detail). To retrieve image characteristics and be able to distinguish between client and imposter images, analyses of the 2D Fourier spectra [106] can be used, with the assistance of DoG in eliminating the noise at the very high frequencies in the Fourier spectra. Besides, DoG helps to reduce possible lighting variations in face images.

Thus, Tan et al. [177] developed two further extensions to the standard sparse logistic regression model, allowing for a quicker and more accurate spoof detection; spare low rank bilinear logistic regression, and nonlinear model via empirical mapping. In particular, they study the differences between two images $I_r(x, y)$ and $I_f(x, y)$; an image of a real live face, and an image of a recaptured face, respectively, under the assumption of the light reflectance of the face surface following *Lambert's cosine law* [133].

The Lambert cosine low is applied to find the intensity of both real and imposter images (I_r and I_f respectively) using the Lambert equation:

$$I(x, y) = f_c(x, y)\rho(x, y)A_{light}\cos\theta, \tag{3.4.3}$$

Where A_light is the intensity of the incoming light, $\rho(x, y)$ is the reflectance coefficient, $\cos = n.s$ is the angle between the surface normal n and the incoming light ray s, and deriving:

$$I(x, y) = f_c(x, y)\rho(x, y)A_{light}(\boldsymbol{n_t.s}), \tag{3.4.4}$$

$$I(x, y) = f_c(x, y)\rho(x, y)A_{light}(\boldsymbol{n_f.s}). \tag{3.4.5}$$

The employed two methods to derive latent samples to be used by the discriminative model:

Variational Retinex-based Method:

The input of the classifier is the illuiminance part of the face image, computed through the minimisation

$$\mu = \arg\min \int_{image} \| \nabla_\mu \|^1 + \lambda |I - \mu| \tag{3.4.6}$$

Where λ is the data fidelity parameter. An estimation of ρ is obtained using Land's Retinex formula through

$$\log(\rho(x,y)) = log(I(x,y)+1) - log(\mu(x,y)+1) \tag{3.4.7}$$

Differences of Gaussian Method:

It is based on the idea that real and imposter images exhibit different local patterns on their Differences of Gaussians. The reason is that imposter images pass through the system's camera twice which leads to missing high frequency details and lower quality.

The work in [177] is very relevant to our study since we are testing the liveness detection algorithm proposed there against our hypothesis that presenting processed imposter images in-front of a face recognition system increases the chances of spoofing a liveness detection system based on this algorithm. Notice that this algorithm has been tested before only against unprocessed imposter images.

3.5 Artificial Neural Networks

Neural networks became an integral part of many machine learning applications in various areas, computer vision being a notable one of these. Artificial Neural networks (ANNs), are connectionist systems representing computations in hierarchical form. ANNs have been inspired by the human neural system and aim at imitating a human brain. See Figure 3.10.

Consider a set of sample data of the form $\mathbf{X} = \{x_1, x_2, ..., x_n\} \in \mathbb{R}^n$, where $\mathbb{R}$ is the set of real numbers and n is the dimension. The associated dataset with binary class labels $y \in 0, 1$ is:

$$\mathbb{D} \triangleq \{\mathbf{X}^{(i)}, y^{(i)})\}|_{i=1}^{N} \tag{3.5.8}$$

where N is the total number of samples in the dataset. Neural networks transform the input X to a desired output using the following non-linear transform:

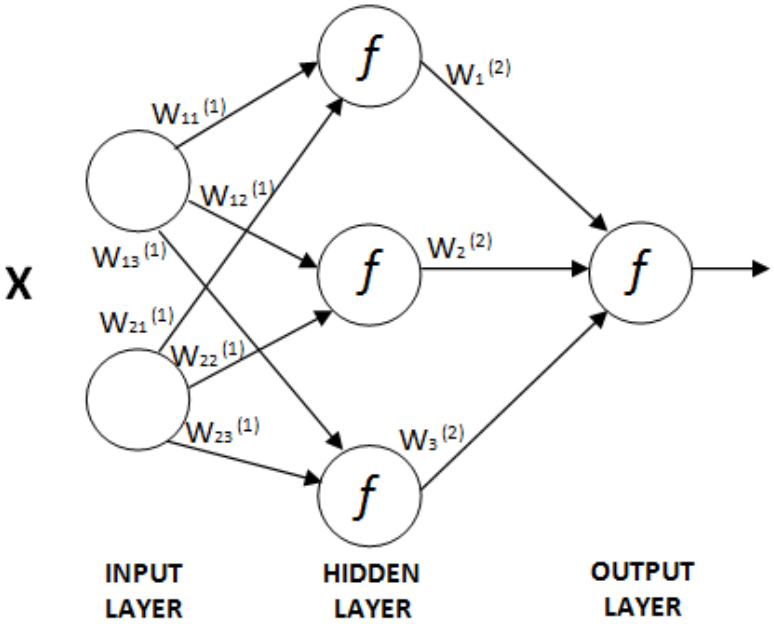


Figure 3.10: General ANN structure.

$$\hat{y}^{(i)} = f_{(li)}(f_{(li-1)}(f_{(li-2)}(\mathbf{X}^{(i)}; W_{(li-2)}); W_{(li-1)}); W_{(li)}) \tag{3.5.9}$$

where, f_{li} is the transfer function for layer li. It can be written in a more general form after the decomposition:

$$f_{li}(\mathbf{X}^{(i)}; W_{li}) = f_{non-lin}(\mathbf{X}^{(i)^T} W_{li}) \tag{3.5.10}$$

$f_{non-lin}$ is a non-linear function, such as *sigmoid()*, or *tanh()*.

3.6 Tools and Softwares

3.6.1 Matlab

Our research mainly deals with graphics and image processing algorithms and techniques. Among many tools, MATLAB, which stands for Matrix Laboratory has been chosen as the software for the implementation part of my thesis. MATLAB has many built in vector and matrix manipulation tools, and also is an excellent tool for solving algebraic equations and doing numerical integration, besides being an extremely powerful tool for dealing with and presenting both 2D and 3D images.

MATLAB has a lot of ready implemented tool boxes for image processing, and signal processing. MATLAB version used for the project was R2015a.

3.6.2 Matlab Packages and Toolboxes

SLEP Package

Sparse Learning with Efficient Projections is a MATLAB package providing several functions for solving a family of sparse learning problems [111]. Tan et al. [177] uses the SLEP package, as it offers fast convergence rates and works well with the large scale data they have. For example, to compute a sparse logistic regression with one single regularization parameter we call the function. LogisticR:

$$[x, c, funVal] = LogisticR(A, y, \lambda, opts) \tag{3.6.11}$$

where, A is the data matrix, where each row of which corresponds to a sample image. y is the labels and is a column vector with a length equal to the number of sample images m, while λ is the regularization parameter, and $opts$ are the advanced options.

The outputs are x, c, and $funVal$; where x is the returned weight vector, c is the intercept, and $funVal$ gives function values during iterations.

Matlab Toolboxes

We also made use of Matlab toolboxes:

- **Image processing toolbox:** perform images processing and analysis, and can speed up algorithm in development. We used this toolbox in our research for image filtering and enhancement, as we produce several filtered images such as sharpened images using the *imsharpen* filter and, and blurry images using the *Gaussian* filter.
- **Signal Processing toolbox:** This toolbox is used for signal processing and analysis. In our research we used the Fast Fourier Transfrom (FFT) to compute the 2D-Fourier of the DoGs of real, imposter and sharpened imposter images.

- **Neural Network toolbox**: in our research, we use this toolbox for creating, training, visualizing, and simulating neural networks.

3.6.3 Evaluation

Liveness detection is subject to two types of errors, either access is granted while the object is an imposter object or access is rejected while the object is a real one. Their rates of these errors are called False Positive Rate (FPR), and False Negative Rate (FNR), respectively. The Receiver Operating Characteristics (ROC) also known as ROC curve, is a graphical representation of the True Positive Rate (TPR) plotted against the Y-axis, while the False Positive Rate (FPR) is plotted against the X-axis, for various values of algorithmic parameters, most commonly thresholds τ.

My research challenge is that when using DoG for deriving latent samples, as in [177], the ROC curve of the performance of the various sparse linear discriminative models shows a very sharp rising from the very beginning, meaning we have a very high TPR values even for every small FPR values. See Figure 3.11. Our research would focus on enhancing the spoofing attacks against such systems, to make the TPR increase slower as the FPR increases.

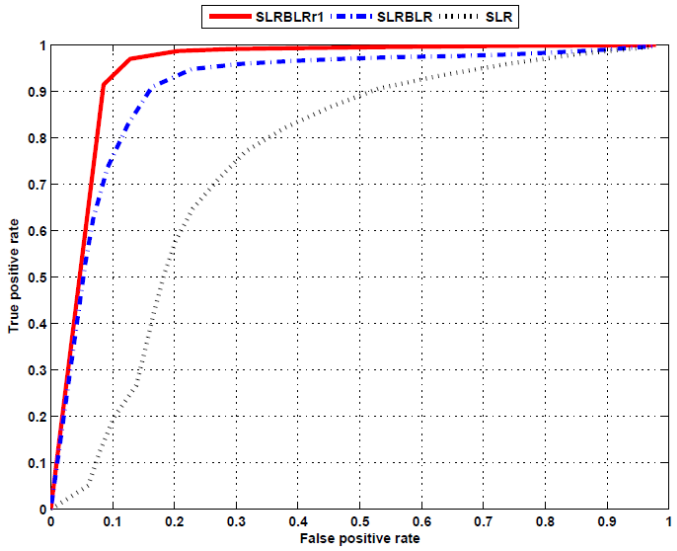


Figure 3.11: ROC curves corresponding to liveness tests with DoG feature images, and various sparse linear discriminative models. Spare Logistic Regression SLR corresponds to the gray dotted curve, Spare Low Rank Bilinear Logistic Regression $SLRBLR$ to the blue dashed curve, and as a specific case of the Sparse Low Rank Bilinear Logistic Regression with rank $= 1$, $SLRBLRr1$ correspond to the red solid curve. [177].

Chapter 4

Evaluating the Resilience of Commercial Face Recognition Systems against Malicious Attacks

This chapter presents an experiment designed to test the resilience of several user verification systems based on face recognition technology against simple identity spoofing methods, such as trying to gain access to the system by using mobile camera shots of the users, their ID cards, or social media photos of them that are available on-line. We also aim at identifying the compression threshold below which a photo can be used to gain access to the system. Four major user verification commercial tools were tested: KeyLemon and Luxand Blink on Windows, and Android Face Unlock and FaceLock on Android. The results show all tested systems to be vulnerable to even very crude attacks, indicating that the technology is not ready yet for adoption in applications, where security rather than user convenience is the main concern.

4.1 Introduction

Biometric authentication systems compare live samples with what the system has previously recorded in its database to insure that only validated people gain access to the system. Currently, biometric identification is used not only in consumer or

business applications, but in the list of users has been extended to include governmental agencies, at high level security applications.

Face recognition is one of the most promising user verification techniques. The ease of using face biometrics comes from the fact that built-in cameras at various resolutions, are available in some of the widely used electronic devices, such as laptops and mobile phones. Hence, many applications have been implemented supporting face recognition for user authentication without the need to use any form of passkeys. Instead, the user is just scanning the face in front of the camera. Recently, developers of smart phones have started including face recognition facility capabilities in their mobile phones. Various vendors opted to include face recognition techniques in their devices, e.g. Apple's iPhone X Face ID; the first mobile phone embedded face recognition system.

Since social media became an essential part of many people's life, millions of images are published on-line on a daily basis. Thus, it is quite likely that among all these heterogeneous sets of uploaded photos there exist clear face images, which can be later exploited by potential attackers to spoof a face recognition system. Figure 4.1 shows an example of an attempt to spoof a face recognition systems using an ID photo or other photos obtained over the Internet.

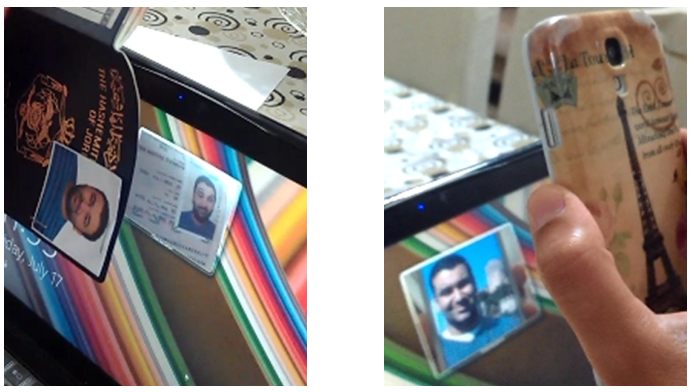


Figure 4.1: Examples of spoofing attacks. **Left:** attempted attack using an ID photo. **Right:** attempted attack using a photo found on Facebook.

Differences between real and recaptured images are not usually subtle substantial, as they both normally look similar with only small difference which the human eyes usually cannot detect.

In a cat-and-mouse game played between attackers and anti-spoofing software and systems developers of successful spoofing attempts have been reported by various groups on several top-branded commercial products. These attacks range from the very basic such as the use of still face images or video sequences, to some more advanced methods, such as using pre-designed 3D masks to bypass a face identification system [108]. On the other hand, Lenovo, Asus, and Toshiba laptops Face Recognition systems were attacked by a team from the University of Hanoi in Vietnam using photos of the real users. Lately, and one week after the release of the new iPhone X, many attackers claimed to be fool-able to "Face ID", which was claimed to be a futuristic new authentication technique. Fingerprint recognition commercial systems also suffer from similar vulnerabilities. For instance, the German hackers group Chaos Computer Club was successful in spoofing the fingerprint scanner in Apples iPhone 5S.

Enhancing the security of face recognition systems is a major challenge since a secure system should be able to withstand a variety of attacks, ranging from systematic algorithmic attacks to attacks based on theft of data. In this chapter, we present an experiment designed to test the resilience of commercial face recognition systems against theft of data attacks. The first part of the experiment uses still face images collected in different ways and from various sources, i.e., instant still images taken during the experiment by a smartphones camera; ID card photos; images found through Google image searches and images on social media platforms such as Facebook and WhatsApp. In the second part of the experiment, we resize some of the still images that were successfully used to gain access to the system and we find the minimum resolution required for such an attack.

The main contribution of the chapter is a demonstration that some of the well-known face recognition systems can be spoofed by crude techniques and images that can be easily found on-line. In practice, that means the development of commercial face recognition system should give more consideration into the possibility that an attacker can use publicly accessible images for attacking their systems. Also, we are highlighting the fact that a lower than the original image resolution, does not necessarily result in a failed spoofing attempt, in fact even very low quality images

can be used to successfully gain access to various commercial systems.

The main limitation of our study is that due to the large number of available commercial face recognition systems, we could not run our tests on the majority of these systems. Furthermore, as it seems that the developers of such systems prioritise user convenience over security, and as of these many products have a variation on possible security configurations, it is difficult to quantitatively measure the potential robustness of the underlaying anti-spoofing algorithms.

The rest of the chapter is organized as follows: In Section 4.2 we review the four tested commercial face recognition systems. In Section 4.3, we discuss our experimental setup, introducing the relevant software and hardware, and describing the environment of the experiment. In Section 4.4, we present and discuss the results, and finally we conclude in Section 4.5.

4.2 Commercial Face Recognition Systems

Face recognition systems are able to identify and verify the identity of a person from a digital image from a stil photo or a short video clip. Currently, there are hundreds of such systems employed various applications such as security or payments. Among these systems, for our study we chose for testing four widely available commercial tools; KeyLemon, Luxand Blink, Android Face Unlock and FaceLock for Apps.

KeyLemon

KeyLemon is a biometric authentication solution based on face and speaker recognition. It offers a non-password login to Windows using face recognition on a standard webcam and it can also be used for login to web based systems such as Facebook and Twitter. KeyLemon claims that its latest face recognition algorithms enhance security by using 3D depth sense cameras to combine depth, near-infrared and colour information [1]. Here, we run the freely available version of the system on a common laptop hardware configuration which did not support such features.

Luxand Blink

Luxand Blink is one of the most popular user verification systems to be used as a convenient alternative to passwords. It supports quick login on different operating systems, e.g. Windows, Mac OS, linux, iOS and Android. Luxand Blink's algorithm processes the coordinates of 40 facial feature points such as mouth corners, nose tip, eyes, eye corners and eyebrows [4].

Android Face UnLock

It was first released for the Android 4.0 "Ice Cream Sandwich" for unlocking Android mobile phones. Afterwards, an enhanced version was offered on Android 4.1 "Jelly Bean" with a new liveness test option embedded, which checks if the person in front of the camera is blinking making sure they are real [2]. Being a non-standard feature requiring user interaction, here we did not enable this liveness test.

FaceLock for Apps

FaceLock for Apps is an alternative face recognition tool for locking either an Android phone or some of its applications. It is a very popular system, having a 4/5 star rating based on the feedback from more than 10,000 users on Google Play store [3].

Windows Hello for Windows 10

Windows Hello is a face and/or fingerprint biometric authentication system providing instant access to Windows 10 devices. It is claimed to be more convenient and secure than the standard typing of password. However, special hardware will be required to use Windows Hello, including a fingerprint reader and/or an illuminated IR sensor for detecting and verifying the liveness of faces [20]. We did not test Windows Hello on Windows 10 because of the extra hardware requirements.

4.3 Experimental Setup

4.3.1 General Setup

Two operating systems were tested in the experiment: Windows 8.1 and Google Android 4.4.2. A laptop with Windows 8.1, Intel Core i3 CPU @ 1.90 GHz, 4 GB RAM, 64-bit Operating System and a Front HD webcam was used to test the KeyLemon and Luxand Blink systems. A Samsung Galaxy S4 mobile phone running Android 4.4.2 with a 2MP front camera was used to test the Android built-in Face Unlock and the FaceLock for Apps. A Samsung Galaxy S4 with a 13 MP rear camera was used to take images of the participant at a resolution of 2322 × 4128 pixels (9:16). The images were taken from different distances: one at short distance (50 cm) where the face was 15% of the full image, one at intermediate distance (100 cm) where the face was 3% of the full image, and one at far distance (150 cm) where the face was 0.8% if the full image.

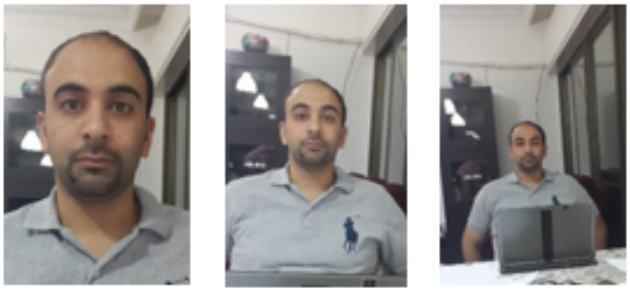


Figure 4.2: Instant photos of a participant taken by a Samsung Galaxy S4 Rear 13 MP camera at various distances. **Left to right,** the camera distance is: (i) 50 cm, (ii) 100 cm, (iii) 150 cm.

The experiment took place in Amman, Jordan in an indoors environment under stable, good illumination conditions. There were five participants in total, of various ages, male and female, without any specific requirement for their computer skills. Since we did not run any statistical tests over the results, there was no need for a larger number of participants. On the other hand, the repetition of essentially the same experiment five times, rather than just one or two, gave us more confidence in our analysis and conclusions.

4.3.2 Systems Setup

The recommended default levels of security were chosen for all tested systems. In particular, the KeyLemon security level, which ultimately is a trade-off between convenience of use and security, was set at medium recognition accuracy. In Luxand Blink, the high convenience mode was set, while the medium security level was chosen for the Facelock for Apps system. The Android Face Unlock does not have any such parameters to set up.

4.3.3 The Experiment

Prior to the experiment each of the five participants was asked to sign a consent form. The form contained a brief about the experiment and a reassurance that there were no direct risks to them by participating to the study. The duration of the experiment was about one hour per participant and it was split in to four sessions.

In the first session, the participants were asked to register with the four systems and test their registration. In the second session, the participants were involved in a photoshoot session in which three frontal face images of them from distances of 50 cm, 100 cm and 150 cm were taken and then the participants were asked to try to gain access to the system using these images displayed on a smartphone camera. In the third session, the participants were asked to try to gain access to the systems using the photo of their Jordanian national ID card. Finally, the participants were asked to find three face-front photos of them published on the Internet/social media and these photos, displayed on a smartphone camera were again used to try to gain access to the systems. After the end of the four sessions, we compressed the participant's photos that were taken in the second session at distance 50 cm from the camera and tried to gain access to the system until the maximum compression ratio still allowing access to the system was found.

4.4 Results

4.4.1 Gaining Access to the System

Tables 4.1, 4.2, and 4.3 show the results from each experimental session and each participant; use of instant photos taken by the mobile phone from various distances (50 cm, 100 cm, and 150 cm), use of ID photos, and use of photos found on the Internet or means in social media sites, respectively.

The participants were all able to gain access to all systems with a smartphone shot taken from a distance of 50 cm, while frontal images taken from distances of 100 cm and 150 cm were not able to gain access to any of the tested systems. However, as the compression results in Section 4.4.2 indicate, the participant can easily gain access using longer distance photos, after zooming into their face and cropping the image. Getting access to the systems using ID photos was partially successful. Android Face Unlock had the highest by-pass rate standing at 3/5, followed by keyLemon with 2/5 and then Luxand Blink and FaceLock for Apps with 1/5. Photos on the Internet and/or social media were also partially successful in gaining access to the tested systems. KeyLemon had a successful by-pass rate of 7/15, followed by Luxand Blink and Android Face Unlock with 5/15 and FaceLock for Apps with 3/15.

4.4.2 Compression Results

Table 4.4 shows the lowest filesize for the compressed photos that were taken at a distance of 50 cm from the camera such that access to the system was still possible. The filesizes are given as percentages of the filesize of the original images. All photos are encoded in JPEG, the resolution is 2322×4128 (9:16) and the face covers approximately 15% of the whole image.

We notice that even highly compressed images successfully can be used to gain access to the tested systems. That suggests that the failure in gaining access with the long range images (100 cm and 150 cm) was most probably due to particular system settings requiring the recognised face to be closer to the camera, rather than the lack of enough information in the long range images. To test this hypothesis, the

Table 4.1: Results of instant photos using a mobile phone at various distances.

Instant Photo by a mobile phone taken from a close distance (approx. 50 cm) was successful in gaining access to the following system: (Yes/No)						
System	**E1**	**E2**	**E3**	**E4**	**E5**	**Average**
KeyLemon	Yes	Yes	Yes	Yes	Yes	**100%**
Luxand Blink	Yes	Yes	Yes	Yes	Yes	**100%**
Android Face Unlock	Yes	Yes	Yes	Yes	Yes	**100%**
FaceLock for Apps	Yes	Yes	Yes	Yes	Yes	**100%**
Instant Photo by a mobile phone taken from a close distance (approx. 100 cm) was successful in gaining access to the following system: (Yes/No)						
System	**E1**	**E2**	**E3**	**E4**	**E5**	**Average**
KeyLemon	No	No	No	No	No	**0%**
Luxand Blink	No	No	No	No	No	**0%**
Android Face Unlock	No	No	No	No	No	**0%**
FaceLock for Apps	No	No	No	No	No	**0%**
Instant Photo by the mobile phone taken from a close distance (approx. 150 cm) was successful in gaining access to the following system: (Yes/No)						
System	**E1**	**E2**	**E3**	**E4**	**E5**	**Average**
KeyLemon	No	No	No	No	No	**0%**
Luxand Blink	No	No	No	No	No	**0%**
Android Face Unlock	No	No	No	No	No	**0%**
FaceLock for Apps	No	No	No	No	No	**0%**

images taken from the 150 cm distance were cropped around the face, compressed as 50 kb JPEG files, and finally resized by an x2 zoom. In all five cases, these cropped, compressed and zoomed-in images were successfully used to gain access to all five systems.

Table 4.2: Results of using ID Photos.

ID photo was successful in gaining access to the following system: (Yes/No)						
System	**E1**	**E2**	**E3**	**E4**	**E5**	**Average**
KeyLemon	No	No	Yes	Yes	No	**40%**
Luxand Blink	No	No	No	Yes	No	**20%**
Android Face Unlock	No	Yes	Yes	Yes	No	**60%**
FaceLock for Apps	No	No	No	Yes	No	**20%**

Table 4.3: Results of using photos on the Internet and social media.

Photos on the Internet/Social Media were successful in gaining access to the following systems: (Number of successful images out of 3)						
System	**E1**	**E2**	**E3**	**E4**	**E5**	**Average**
KeyLemon	1	1	2	2	1	**46.67%**
Luxand Blink	0	1	1	2	1	**33.33%**
Android Face Unlock	1	0	1	2	1	**33.33%**
FaceLock for Apps	0	0	0	2	1	**20.00%**

Table 4.4: Smallest filesize of compressed images as a percentage of the original.

Tool	**E1**	**E2**	**E3**	**E4**	**E5**	**Average**
KeyLemon	2%	3%	2%	2%	3%	**2%**
Luxand Blink	4%	4%	4%	5%	4%	**4%**
Face Unlock	1%	1%	2%	1%	1%	**1%**
FaceLock for Apps	2%	2%	3%	2%	2%	**2%**

4.4.3 Processing the Images with Noise and Blur

Since face recognition algorithms tolerate a lot of noise, further experiments were conducted trying to gain access after using some simple image processing techniques on our images: addition of "salt and pepper" noise and Gaussian blur. The addition of noise and the removal of any pre-existing noise through the Gaussian blur can

be seen as simulations of a stochastic resonance phenomenon, and thus, this part of the experiment can be seen as an evaluation of the robustness of the tested systems under stochastic resonance.

Tables 4.5 and 4.6, show the results of access/denied access for the keyLemon system after adding various amounts of salt and pepper noise and a Gaussian blur with different sigma values.

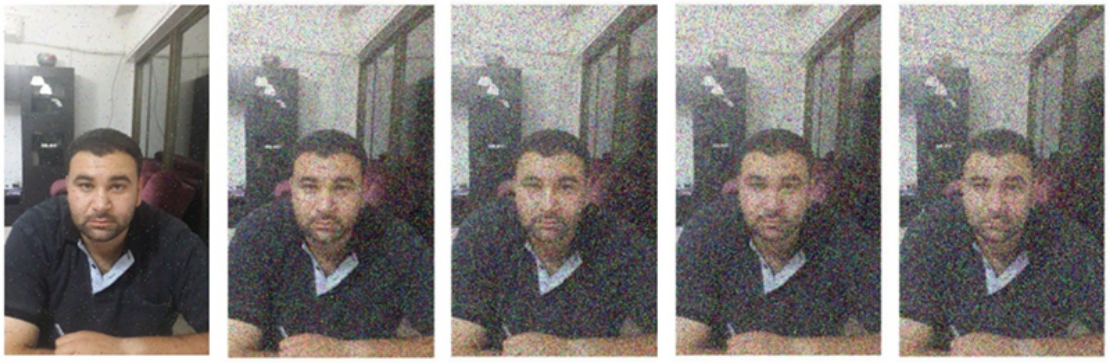


Figure 4.3: Sample photos with additional salt and pepper noise of various amounts. **Left to right:** 0.01, 0.2, 0.28, 0.29, and 0.35.

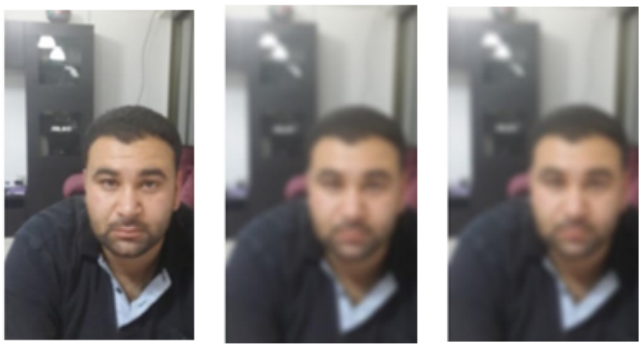


Figure 4.4: Sample photos after applying a Gaussian filter with different σ values. **Left to right:** 10, 27, and 28.

4.5 Conclusions

We tested four user verification systems based on face recognition against basic, direct presentation attacks. We found all of them to be vulnerable even against

Table 4.5: Access/denial results of the keyLemon system after applying salt and pepper noise of various amounts.

Density	0.01	0.2	0.28	0.29	0.35
Result	successful	successful	successful	unsuccessful	unsuccessful

Table 4.6: Access/denial results of the keyLemon system after applying Gaussian filter with different sigma values.

Sigma (σ)	10	27	28
Result	successful	successful	unsuccessful

the crudest of attacks, such as using highly compressed still images taken with smartphone cameras, or using still images that can be found on social media. We believe that part of the high success rate of the attacks is due to the developers of the systems prioritising user convenience over security, at least in the default configuration of their products. However, the question on whether face recognition, at least by its own, is suitable for user verification can also be raised. Indeed, face recognition seems to rely on data which are personal in nature but, nevertheless, are often either already in the public domain, or can be easily stolen.

Chapter 5

Resilience of Luminance based Liveness Tests under Attacks with Processed Imposter Images

Face recognition systems are widely used for user authentication in common everyday applications. However, their vulnerability to even crude imposter photo attacks, as for example when an imposter gain access to a system by holding a photo of the rightful user in front of the camera, means that their use is restricted to applications where security is not considered critical. Liveness tests are countermeasures against such attacks, aiming at verifying that a live face rather than a photo stands in front of the camera of the face recognition system.

The current state of the art, such as the Tan et al. paper [177], is based on machine learning algorithms trained to distinguish between images of live faces taken by the face recognition system and images of photos fed to the system by the imposters. In particular, it has been established that different reflectance properties of these two categories of surfaces, i.e., real face and photo, can lead to the development of effective liveness test.

In this chapter, we study the resilience of this standard liveness test against imposter photo attacks, under the additional assumption that the photos used in the attack may have been processed. In particular, we study experimentally the effect of common image processing operations such as sharpening and smoothing,

as well as corruption with salt and pepper noise. The results verify and quantify the claim that this type of liveness tests rely on the fact that the imposter photo images are usually less sharp than live images of faces. We argue that this indicates a possible vulnerability of such liveness tests from attacks with processed imposter images.

5.1 Introduction

In this chapter we explore potential vulnerabilities of the liveness test proposed in the Tan et al. [177] by studying the effect on its performance of simple image processing operations applied of the imposter images, such as sharpening and smoothing. We focus on the variant of their algorithm which is using differences of Gaussians to create the input data and sparse logistic regression to create the classifier.

As a possible explanation of why their classifier is effective, Tan et al. observe that images of face photos fed into the system by imposters tend to be smoother, as in the scene that they lack in details. In this chapter, we verify and quantify this claim by processing the imposter images of the NUAA database and measuring the performance of their algorithm. We note that, as expected, the sharpening of the imposter images reduces the accuracy rates of their liveness test, while the smoothing of the imposter images increases accuracy rates. We also note that this is a demonstration of a potential vulnerability of their liveness test. Indeed, one can reasonably expect that by holding a sharpened face photo in front of the system's camera, the image read by that camera will also be sharper. That is, one can reasonably expect that the sharpening attack we demonstrated on the imposter images of the NUAA database can be replicated under real life conditions.

The main contribution of the chapter is a demonstration that the accuracy rates of the liveness tests in [177] are sensitive to the processing of the imposter images. In practice, that means that a rigorous evaluation of a liveness test against imposter image attacks should take into account the possibility that the attacker has processed the images they used to gain access to the system. We believe that this is a rather simple attack enhancing technique which has been largely overlooked in the literature

of luminance based liveness tests.

The main limitation of our study is that we process the imposter images of the NUAA database, which are images of a photo of the subject, rather than processing the photos of the subject and recapturing them with the camera before feeding them into the system. While this allows us a better and more quantitative understanding of the basic principle underlying the Tan et al. algorithm, we note that it is not a direct attack and that the effect of a direct attack, consisting of processing the photo of the subject and feeding it into the system, will be studied in Chapter 6.

The rest of the chapter is organized as follows: In Section 5.2, we briefly discuss the design of our experiment. In Section 5.3, we present the results and we briefly conclude in Section 5.4.

5.2 Implementation

In this section, we first discuss some details of the liveness test of Tan et al. and our implementation of it and then we describe the design of our experiment.

5.2.1 Liveness Test

In [177], Tan et al. proposed a series of liveness tests based on the same principle. Information sensitive to the reflectance properties of the scene is extracted from the image and it is used to train a binary classifier so that it can distinguish between images of live faces and images of photos of faces. The variant we implemented here extracts a Difference of Gaussians from the image and uses it to train a sparse logistic regression classifier.

Regarding the difference of Gaussians of the images, following the recommendation in [177], we smooth the image using a Gaussian filter with $\sigma_1 = 0.5$ and a Gaussian filter with $\sigma_2 = 1.0$ and then compute the difference of the two smoothed images.

Regarding the machine learning part of the algorithm, following the notation and parameter choices in [177], we use the class labels $\{-1, 1\}$, where -1 corresponds to client images and 1 to imposter images and the conditional probability of the

imposter class $y = 1$ is given by:

$$\text{Prob}(y|x) = \frac{1}{1 + \exp(-y(w^T x + b)} \tag{5.2.1}$$

where x is the sample image, and w and b are the weight vector and the intercept returned by the logistic regression. To avoid overfitting, the values of w and b are computed through the minimization of the cost function

$$\min_{w,b} \ \text{loss}(w, b) + \lambda \|w\|_1 \tag{5.2.2}$$

where λ is a user defined constant favoring sparse weight vectors, that is, vectors with most of their elements equal to zero, and loss is the standard loss function of the logistic regression

$$\text{loss}(w, b) = \frac{1}{m} \sum_{i=1}^{m} \log(1 + \exp(-y_i(w^T x_i + b))) \tag{5.2.3}$$

where m is the size of the training set of images x_i with associated labels y_i.

The choice of λ can have a significant effect in the performance of the algorithm and might depend on the size of the training set. In our implementation, using a training set of 1000 images we found experimentally that $\lambda = 0.25$ gives good results. Figure 5.1 shows the ROC curves of the liveness test for several values of λ. We notice that the ROC curves corresponding to $\lambda = 0.2, 0.3$ are generally higher than the others, indicating that the algorithm performs well for values of λ in the range [0.2, 0.3].

Following [177], we also used Matlab as our implementation platform and the SLEP package in particular for the sparse logistic regression.

5.2.2 Experimental Design

We used images from the open access NUAA photograph imposter database, see Section 3.2 for more details. The database was constructed using a low cost camera and contains photos of 15 different subjects in various poses taken under different illumination conditions. The images are organized into the two categories: the *client images* which are images of live faces, and the *imposter images* which are images

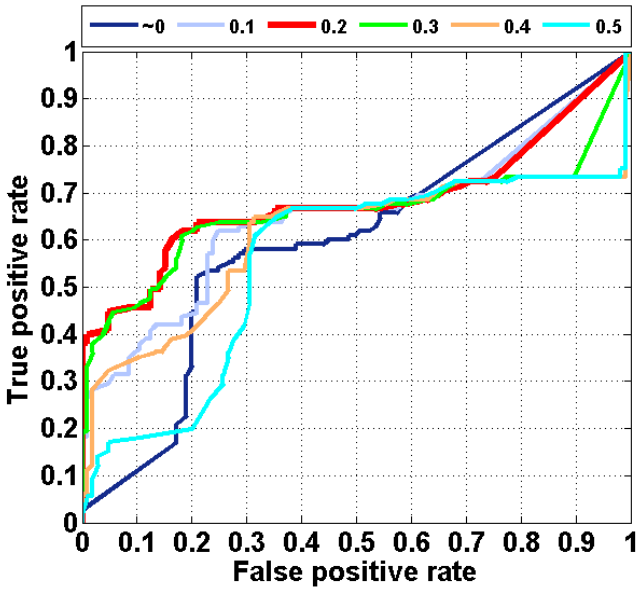


Figure 5.1: ROC curves for several values of λ. The algorithm performs best for values of λ between 0.2 and 0.3.

of photos of the subjects. The size of all images is 64×64 and they are grayscale encoded in RGB.

Our training dataset consisted of a total of 1000 client and imposter images. Our test set consisted of several subsets, each one containing 105 images, i.e., seven from each subject. More specifically, we used subsets of:

(i) client images

(ii) imposter images without any alteration

(iii) imposter images sharpened by subtracting from them the response of the Laplacian filter

(iv) the sharpened images in (iii) blurred by a Gaussian filter with $\sigma = 1.25$

(v) four subsets of imposter images sharpened with the *imsharpen* Matlab function, which subtracts from the images a blurred version of it, for parameter values of 0.5, 1.0, 1.5 and 2.0, respectively, giving different amounts of sharpening

(vi) four subsets of sharpened images blurred by a Gaussian filter with $\sigma = 0.1$, 0.5, 1.25 and 2.0, respectively

(vii) four subsets of images with different amount of salt and pepper noise at 0.01, 0.1, 0.5 and 0.9, respectively

Figure 5.2 shows typical examples of test set images.

Figure 5.2: Test images. **Left to right:** (i) client, (ii) imposter, (iii) sharpened imposter, (iv) sharpened and blurred imposter, (v) imposter with salt and pepper noise added to it.

5.3 Results

Figure 5.4 shows the ROC curves of the original liveness test and after the imposter test images have been sharpened by subtracting from them the response of the Laplacian filter, or processed by a Laplacian filter followed by Gaussian blurring. We notice that, as expected, the performance of the liveness test decreases when the imposter images have been sharpened even with the very basic sharpening algorithm we used. On the other hand, again as expected, the performance of the liveness test increases when the imposter images are smoothed, demonstrating that the sharpness of the image is a key factor in distinguishing between client and imposter images by the Tan et al. algorithm [177].

In the next experiment we want to establish that the changes in the performance of the liveness test is commensurable with the amount of sharpening and blurring applied to the imposter images. Figure 5.3 shows various the ROC curves when the imposter images are sharpened using the *imsharpen* Matlab command. The strength of the *imsharpen* command is controlled by a user defined parameter and we tried

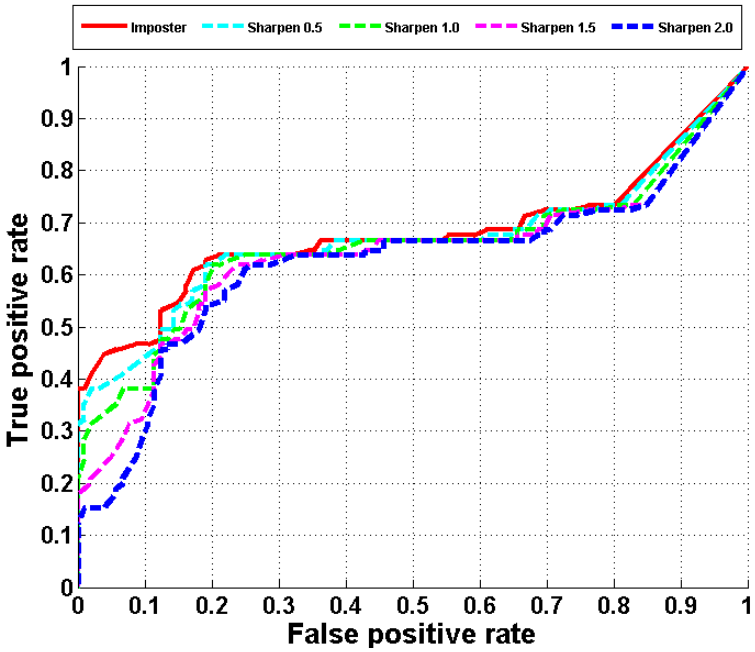


Figure 5.3: ROC curves for the liveness test with different amounts of sharpening applied on the imposter images.

the values 0.5, 1.0, 1.5 and 2.0. We notice that larger amounts of sharpening on the imposter images result into larger decreases in the performance of the liveness test. Similarly, in Figure 5.5 we show the ROC curves when the Laplacian filter sharpened imposter images are blurred with Gaussian filters with $\sigma = 0.1$, 0.5, 1.25 and 2.0, respectively. As expected, we notice that larger amounts of smoothing result to larger increases in the performance of the liveness test.

Finally, we experimented with the addition of various amounts of salt and pepper noise on the imposter images. This test is relevant in the context of liveness tests, since in [130] it was shown that commercial face recognition systems can cope with large amounts of salt and pepper noise but, as a consequence they are also vulnerable to imposter image attacks even when the imposter images contain large amounts of salt and pepper noise. Figure 5.6 shows the results when salt and pepper noise with probabilities 0.01, 0.1, 0.5 and 0.9, respectively, was added to the imposter images. We notice that the addition of noise increases the performance of the liveness test and that the performance gain is commensurable with the amount of added noise.

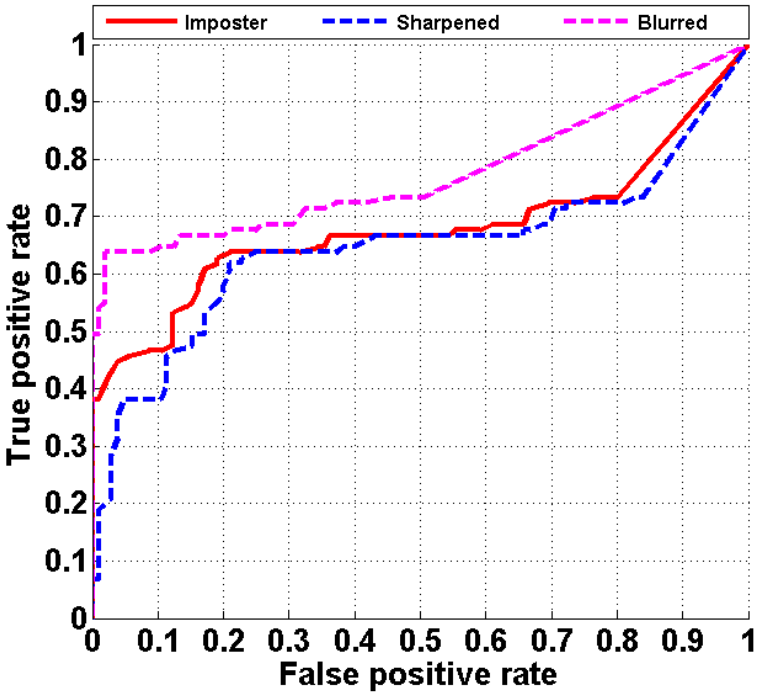


Figure 5.4: ROC curves for the liveness test when sharpening the imposter with the Laplacian filter, and then blurring them. The value of λ here is 0.25, with Gaussian blur $\sigma = 1.25$.

5.4 Conclusions

When evaluating a liveness test against imposter image attacks, the assumption that the imposter photos are used by the attacker unprocessed is rather optimistic. Indeed, in real life situations, we should expect that an attacker will process an imposter photo before using it, as long as they know that would increase their chances of successfully evading a liveness test.

Motivated by such an observation, we evaluated the resilience of a standard luminance based liveness test in conjunction with certain image processing operations on the imposter database. Our results verified and quantified the assumption that luminance based tests rely on the different amount of sharpness between images of live faces and imposter images. In particular, as expected, we found that the sharpening of the imposter images decreases the accuracy of the liveness test while smoothing or sharpening followed by smoothing increases the accuracy rates.

While we reasonably expect that by presenting a sharper printed photo in front

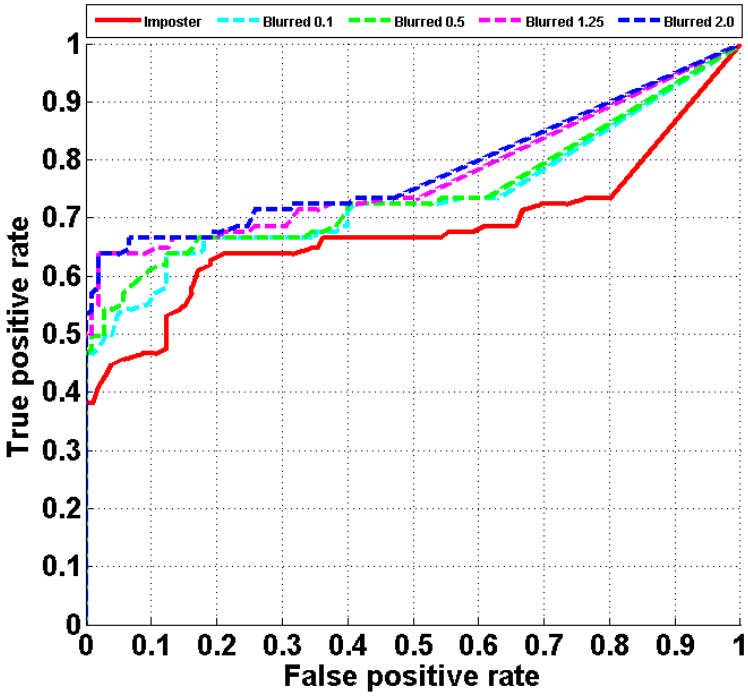


Figure 5.5: ROC curves for the liveness test with different amounts of Gaussian blur on the imposter images.

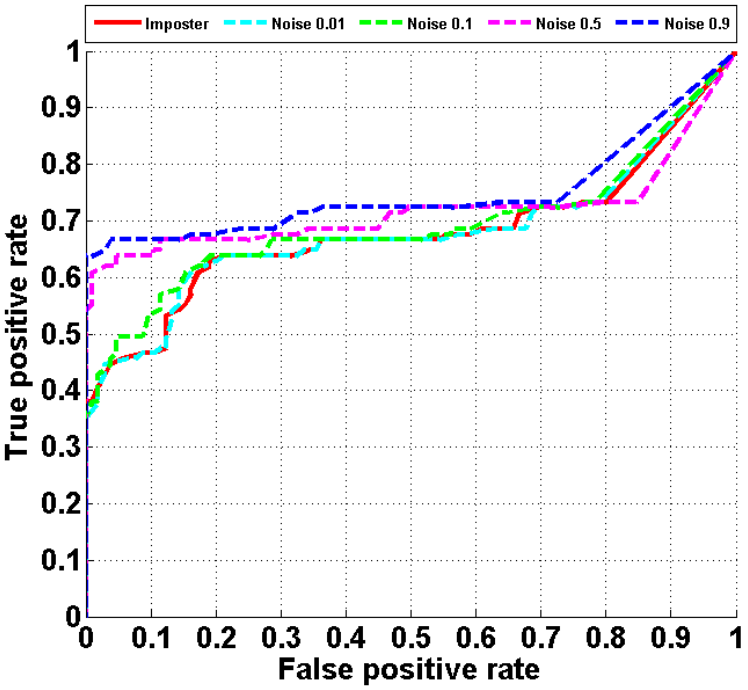


Figure 5.6: ROC curves for the liveness test with different amounts of salt and pepper noise added to the imposter images.

of the camera will result into a sharper image acquired by the camera, this is an assumption that still needs to be verified. Thus, in chapter 6 we will simulate and evaluate imposter image attacks that use processed images under realistic conditions. That is, by printing or displaying on a tablet screen sharpened imposter images and evaluating the resilience of the luminance based liveness test.

Chapter 6

Designing a Facial Spoofing Database for Processed Image Attacks

Face recognition systems are used for user authentication in everyday applications such as logging into a laptop or smartphone without the need to memorize a password. However, they are still vulnerable to spoofing attacks, as for example when an imposter gains access to a system by holding a printed photo of the rightful user in front of the camera. In this chapter we are concerned with the design of face image databases for evaluating the performance of anti-spoofing algorithms against such attacks.

We present a new database called DURHAM FACE Database, supporting testing against an enhancement of the standard still photo attack, in that the imposter processes the stolen image before printing it on paper or displaying it on screen. By testing a standard anti-spoofing algorithm on the new database we show a significant decrease in its performance and, as a simple remedy to this problem, we propose the inclusion of processed imposter images into the training set. Part of the research described in this chapter has been published in [131].

6.1 Introduction

The performance of anti-spoofing algorithms is evaluated on databases containing both photos of real people called *client images*, and photos of imposters, which essentially are photos of client images and are called *imposter images*. The design of such a database is a particularly challenging task given the multiple sources of variation in spoofing attacks. Indeed, a whole range of choices, from the choice between a paper photo and an electronic display for the attack, to the type of paper and printer used to print a photo, to the size of that photo and the way it is held in front of the camera, all these factors can impact the effectiveness of the attack and thus the perceived performance of the anti-spoofing algorithm.

The liveness test we used to evaluate the DURHAM FACE Database was proposed by Tan et al. in [177]. There, they proposed several variants of the basic algorithm and the one we chose here trains a sparse logistic regression classifier with the difference of Gaussians of the database images. In a further improvement to the Tan et al. algorithms, [141] apply contrast-limited adaptive histogram equalization before computing differences of Gaussians, increasing the robustness of the test under bad illumination conditions. Differences of Gaussians are also used in [208], where a Support Vector Machine is trained.

The idea that a still image attack can be enhanced by digitally processing the face image before printing it had been suggested in [132]. However, as there are no publicly available face image databases containing processed imposter images, i.e., designed as in Figure 6.1(C), we could not test our assumption. Instead, we used the publicly available NUAA database, measured the performance of the liveness test in [177] against digitally sharpened versions of imposter images, i.e., as in Figure 6.1(B), and argued that the drop in the performance of the liveness test is evidence supporting our assumption.

In this chapter we address a gap in the current practice of the anti-spoofing algorithm evaluation, namely the assumption that the attacker prints the photo of the rightful user as it is, i.e., without attempting to process it in order to increase the effectiveness of the attack. We created a face image database which besides the usual imposter images it also contains imposter images obtained by photo-shooting

printouts of sharpened client images, see Figure 6.1. We tested the database on a standard liveness test [177], and found that the more sophisticated attack which use processed imposter images is more likely to evade detection.

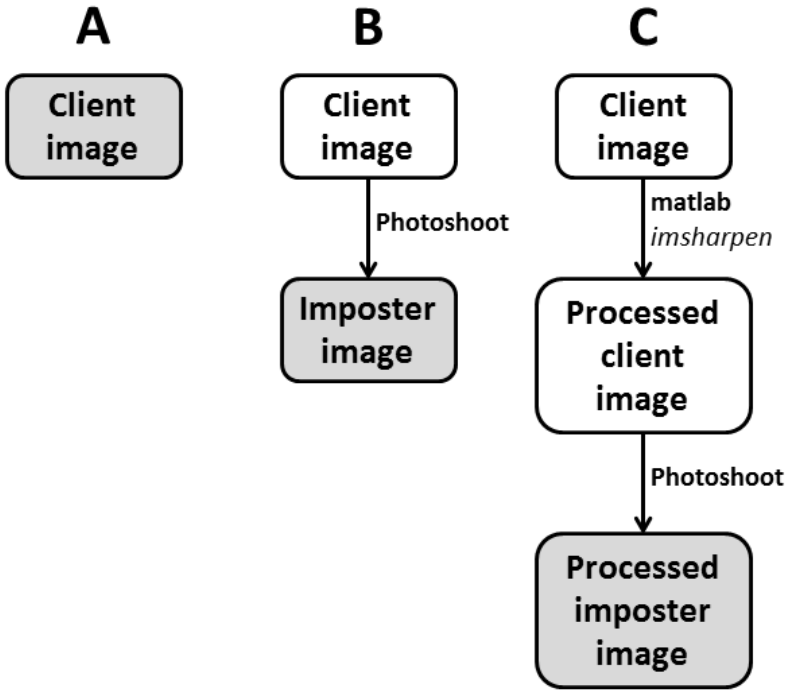


Figure 6.1: The standard database design consisting of client images (A) and imposter images (B), is augmented with processed imposter images (C).

Contribution: the main contribution of the chapter is the design and the construction of a database of face images for testing anti-spoofing algorithms, which, to the best of our knowledge, is the first one based on the assumption that the imposters may use image processing tools to enhance the effectiveness of their attack.

Limitation: as the main limitation of the chapter we note that our current database only serves as a proof of concept, considering only one image processing operation on the client images before they are printed. However, extending the database with imposter images that have undergone other types of processing is a relatively straightforward, even though laborious, process.

The rest of the chapter is organised as follows: In Section 6.2 we briefly discuss the motivation behind the design of the DURHAM FACE database. A detailed description of the database design and the parameters we considered before implementing it are discussed in Section 6.3. In Section 6.4 we present the details of the

DURHAM FACE database. In Section 6.5 we test the database by running on it a standard liveness test and discuss the relevance of the results. A further extension to our database is discussed in Section 6.6. Recommendations for good practices in designing a face database for liveness test evaluation based on our experience from our research are summarized in Section 6.7. We conclude in Section 6.8.

6.2 Motivation for Designing a New Database

The design of a database for evaluating liveness tests is a challenging task, given the variability of form of all possible imposter attacks. An additional difficulty is that the performance of the current state-of-the-art liveness tests seems to drop significantly when the imposter attack deviates even slightly from the protocol that was used to produce the imposter samples of the training set. For example, when a different paper type is used to print imposter images, or a different printer or electronic display, or a different camera for recapturing imposter images. A similar problem has been observed in the behaviour of state-of-the-art algorithms for the classic face recognition problem, where it is usually referred to as the *interoperability* problem, see for example Gallbaly and Satta [60]. As a result, it is important to have an exact description of the protocol under which a face image database was constructed, even if that contains a number of tedious and seemingly irrelevant details.

In Section 3.2, we reviewed some of the publicly available databases of facial spoofing sets such as the NUAA database, which as the DURHAM FACE Database consists of still images, and three databases containing short video sequences, which nevertheless can also be used to evaluate still image liveness tests after extracting frames from the video sequences. We note that none of these databases contains images or videos produced by processed imposter images, i.e., as in Figure 6.1(C).

In real life, it is unlikely that an attacker will use unprocessed images if they know that some simple processing will increase the effectiveness of their attack. In [131] the resilience of a standard luminance based liveness detection test was evaluated against processed imposter attacks. The assumption was that when imposters smoothed,

the accuracy of the liveness test increases, while in contrast, the sharpening of imposters decreases the accuracy of the test. The NUAA database was used to test that assumption, but since it does not contain any processed imposters, we just digitally smoothed and sharpened imposter images, instead of smoothing and sharpening client images and then recapturing them.

6.3 Database Design and Parameter Fine Tuning

As already mentioned, designing a facial spoofing database is far from easy and straightforward. There are plenty of challenging issues and parameters to be considered and controlled by the developers to achieve the optimal database behaviour. In this section, we discuss some of the parameters we considered for our design with a brief description of pilot tests we did for each considered parameter.

Before deciding on the choice of camera, a pilot shooting session was conducted at the Imaging Laboratory of Durham University with 3 different cameras mounted to tripods; a professional Canon EOS Rebel T3i (600D) with a 18-250mm lens, on iPhone 6S mobile camera, and a commodity webcamera. Various parameters were taken into consideration, such as illumination conditions, camera focus mechanism (i.e., auto or manual focus), the distance between the camera lens and the object, captured image size. Due to the large number of parameters, interactions between them where not considered and we studied the effort of each individual parameter separately.

Several pilots were designed to obtain a better understanding of each parameter and take the right decisions for creating an optimally designed database. First we had a photo-shooting session with our three available cameras, and we noticed that when using a mobile screen for displaying client images, the produced imposter images were slightly blurred due to being out of focus, see Figure 1.3. Thus, we decided to evaluate two focus mechanisms; manual and auto focus.

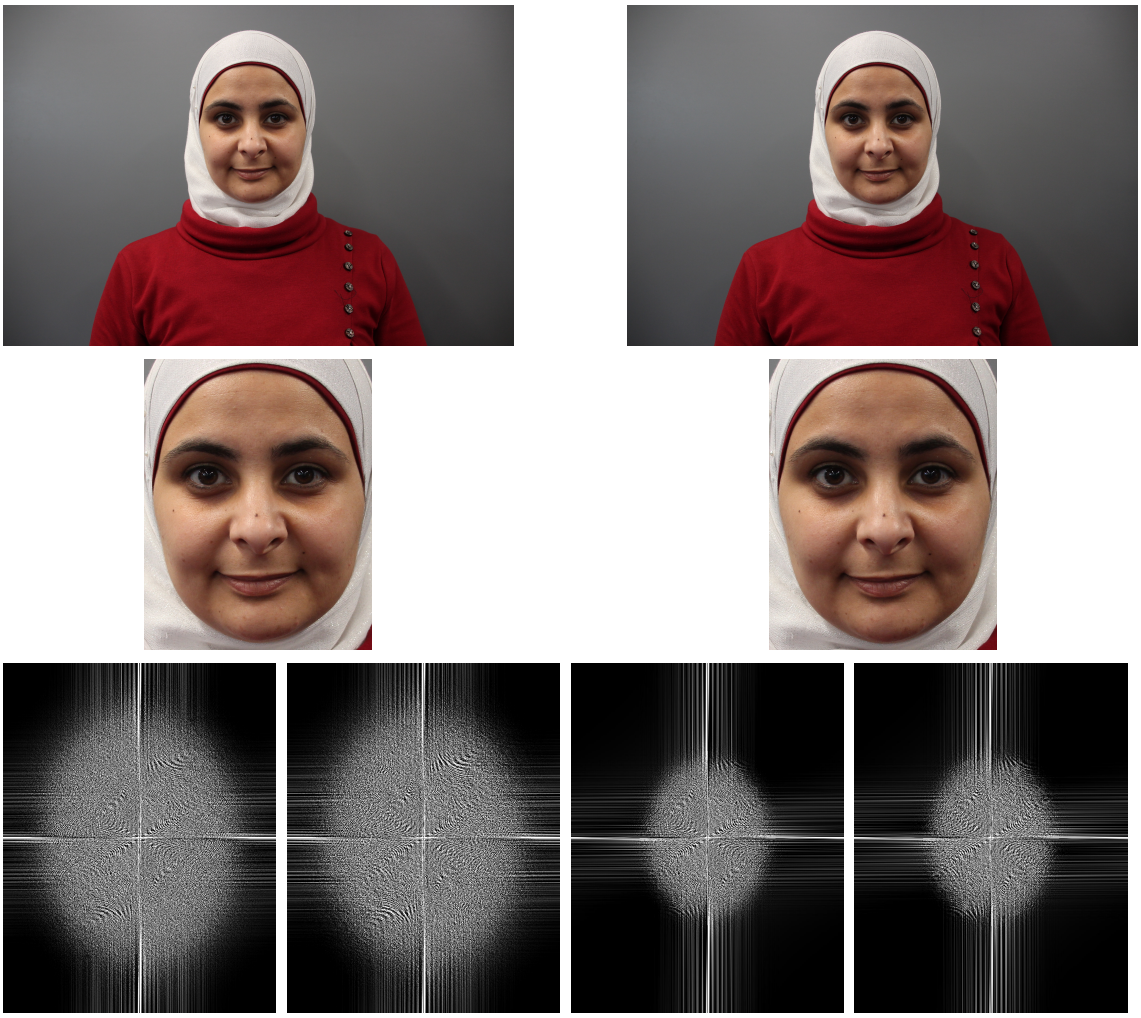


Figure 6.2: Client images captured by both manual and auto-focus mechanism. **Row 1:** Original client images using: Left: auto-focus, Right: manual-focus. **Row 2:** Cropped images of Row 1. **Row 3:** The 2D Fourier transform of the DoG filtered images with sigma values of: (left to right): $\sigma = 1.0$ & 2.0 and $\sigma = 2.0$ & 4.0 for auto and manual images.

Focus Mechanism

Before opting for using the auto-focus function of the camera we compared imposter images captured with manual-focus and images captured with auto-focus and found that the latter are more likely to be sharper, and thus, more challenging to classify correctly. In Figure 6.2, a sample of two face images taken using auto and manual-focus mechanism, are shown as well as the Fourier of the difference of Gaussians for both images. We see that the face captured by the auto-focus mechanism has a richer horizontal component of high frequency areas compared to the manual-focus image. We also noticed that the auto-focus mechanism effectively prevents the capture of very blurry out-of-focus images by blocking the shutter release.

Focus mechanism, camera type, image size and distance from camera

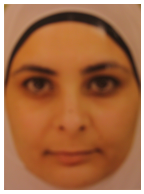
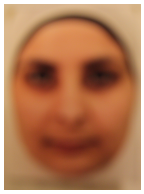
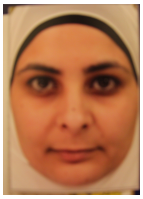
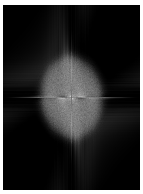
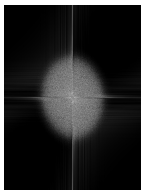
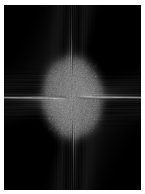
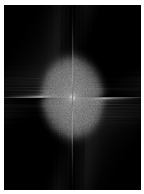
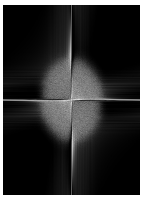
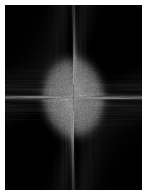
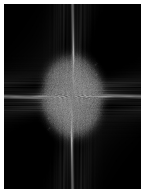
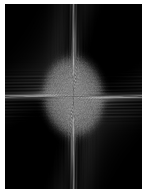
Table 6.1, shows some imposter images of various sizes, captured from printed papers of different material, using a professional camera with both manual-focus and auto-focus. To keep the size of the recapture image constant, we had to bring smaller size images nearer to the camera, approximately 10 cm away from the lens in the passport size case. As a result, the image was out of the depth of focus and no photos could be taken from that of closer distances. We decided to recapture only client images of larger size from a reasonable distance from the camera to produce usable imposter images.

Upon the decision of working using auto-focus which is a widely available option for the majority of commercial cameras, we next assessed the type of camera to be used. Our three options were professional camera, mobile phone, and webcam camera, their resolutions ranging from high to low. In the DURHAM FACE database photo-shooting sessions photos were captured using the three considered cameras. Figures 6.10 and 6.11 show original and face-cropped samples from the conducted photo-shooting sessions.

Creating imposter images using paper printed or screen displayed photos can be done either with actual size images at the same distance from the camera as the real image was in during the creation of the client image, or smaller than the original images size and bringing them nearer to the camera.

Thus, we conducted a follow-up brief, using the auto-focus mechanism on the

Table 6.1: Imposters of different sizes captured on various printed paper material by a professional camera, whether with manual-focus, or the auto-focus mechanism.

Manual-focus mechanism					
A4 paper			**Photo paper**		
Passport size	**Medium size**	**Full size**	**Passport size**	**Medium size**	**Full size**
					
					
Auto-focus mechanism					
A4 paper			**Photo paper**		
Passport size	**Medium size**	**Full size**	**Passport size**	**Medium size**	**Full size**
X	X		X	X	
X	X		X	X	

professional and mobile camera, while the web-camera is a fixed-focus camera . The aim was to evaluate the performance of each camera on imposter and sharpened imposter images created by using different amounts of sharpening {2.0, 4.0, 8.0 and 16.0}. Three photo-shooting sessions were conducted for each imposter and sharpened imposter image at passport size, half-size photo size, and full-size. For the consistency of the results, we were careful in having a uniform scale of recapture image size, by making the distance from the camera inversely proportional to the

size of the image. Hence, the distances of the printed or displayed client images from the camera were one, two and four, respectively.

To evaluate the effect of the studied parameters, we are comparing the recaptured images using the Discrete Fourier Spectrum, and measure their complexity by computing the ℓ_1-*norm* [53], see Table 6.2. The imposter image with the most complex signal is considered the more challenging, as real images generally have higher complexity compared to imposters, which are smoother as they lose details when recaptured. As shown in the table, the image captured by the professional camera at four feet distance shows the best average results.

Table 6.2: Imposters and sharpened imposters with {2.0, 4.0, 8.0 and 16.0} amount of sharpening of different sizes {full, half, and passport} and distances from the camera {1, 2, and 4 feet} captured by professional, mobile and web cameras.

Passport Sized Photo and 1 Foot camera distance																		
Professional Camera						**Mobile Camera**						**Webcam**						
0.0	**2.0**	**4.0**	**8.0**	**10.0**	**20.0**	**0.0**	**2.0**	**4.0**	**8.0**	**10.0**	**20.0**	**0.0**	**2.0**	**4.0**	**8.0**	**10.0**	**20.0**	
1.67	1.76	1.73	1.69	2.16	1.90	1.51	1.77	1.59	1.77	2.07	2.34	1.73	1.77	1.41	1.99	1.45	1.40	
1.71	1.62	1.80	1.80	2.07	1.77	1.87	1.43	1.99	1.91	1.88	2.26	1.83	1.72	1.35	1.58	1.23	1.43	
1.76	1.58	1.90	1.74	1.80	1.64	1.51	1.41	1.91	1.88	1.98	2.27	1.78	1.71	1.34	1.38	1.20	1.36	
1.71	1.65	1.81	1.74	2.01	1.77	1.63	1.54	1.83	1.86	1.98	2.29	1.78	1.73	1.37	1.65	1.30	1.40	
Half sized photo and 2 feet camera distance																		
Professional Camera						**Mobile Camera**						**Webcam**						
0.0	**2.0**	**4.0**	**8.0**	**10.0**	**20.0**	**0.0**	**2.0**	**4.0**	**8.0**	**10.0**	**20.0**	**0.0**	**2.0**	**4.0**	**8.0**	**10.0**	**20.0**	
1.76	1.36	1.54	1.82	1.75	2.27	1.42	1.40	1.65	1.98	1.84	1.96	1.95	1.67	1.64	1.72	1.90	1.97	
1.45	1.45	1.50	1.91	2.07	2.43	1.52	1.58	1.81	1.67	2.02	2.08	1.71	1.42	1.45	1.50	1.80	1.56	
1.40	1.40	1.63	0.57	2.10	2.19	1.54	1.69	1.75	1.69	1.85	2.08	1.60	1.85	1.67	1.85	1.58	1.62	
1.53	1.40	1.56	1.44	1.97	2.30	1.49	1.56	1.73	1.78	1.91	2.04	1.65	1.63	1.56	1.67	1.69	1.59	
Full sized photo and 4 feet camera distance																		
Professional Camera						**Mobile Camera**						**Webcam**						
0.0	**2.0**	**4.0**	**8.0**	**10.0**	**20.0**	**0.0**	**2.0**	**4.0**	**8.0**	**10.0**	**20.0**	**0.0**	**2.0**	**4.0**	**8.0**	**10.0**	**20.0**	
1.93	1.87	2.14	2.46	2.51	2.85	2.10	1.81	2.12	2.09	2.23	1.81	2.25	2.15	1.77	2.06	1.75	2.17	
1.92	1.98	1.85	2.15	2.41	2.70	1.89	1.95	2.09	2.16	2.22	1.95	1.94	1.76	1.96	2.07	1.83	2.08	
1.69	1.73	1.81	2.28	1.55	2.80	2.02	2.00	1.87	2.16	2.24	2.00	2.11	2.17	1.81	2.01	1.82	2.14	
1.85	1.86	1.93	2.30	2.16	2.78	2.00	1.92	2.03	2.13	2.23	1.92	2.02	1.96	1.89	2.04	1.82	2.11	

Illumination Conditions

Illumination is one of the most important factors to be considered when creating a spoofing database, since differences in lightning conditions might be affecting the properties of the images. Thus, we consider capturing client images with various lightning conditions, including natural and artificial lights. Figure 6.3 shows an

example of different client images for one object taken under different illumination conditions. These photos were afterwards recaptured to produce imposters.

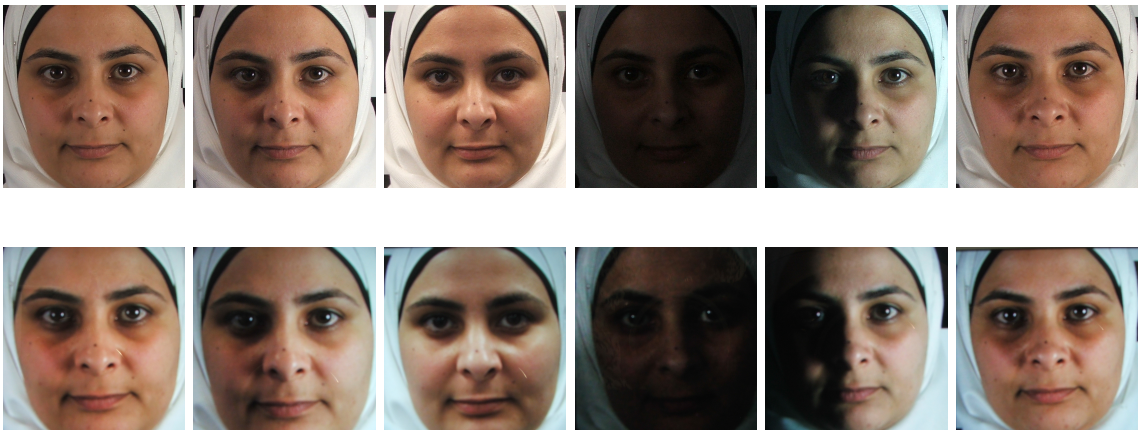


Figure 6.3: Sample of a pilot session for a client under controlled lightning conditions with lights from different angels, and photos of these client images under a certain lightning conditions where the daylight along with the room lights is illuminating the scene. **Left to right:** room lights on and no additional sources of light, with one source of light from left and room lights on, with two additional sources of light from both right and left directions, with day light only, with one source of light from left and day light only, and with two sources of light from both right and left directions and day light only.

Other Parameters to Consider

Image format is also an issue to consider when designing a good database. Since our chosen camera does not provide a variety of options, we chose the only supported format, the JPEG become the default image format of our database. However, the client images are cropped, they will be saved in BMP format which is a lossless image encoding format.

Pilot

Before committing to the development of the full database, we conducted a pilot. We used the *t-test* statistical method to find whether there is any statistical significance

in the differences of the means of imposter images processed with different amounts of sharpening, taken with a professional, mobile, and web camera.

Table 6.3: *t-test*: paired two sample for means.

Mobile and Professional Camera												
Client			**Sharp 2.0**		**Sharp 4.0**		**Sharp 8.0**		**Sharp 10.0**		**Sharp 20.0**	
	V1	**V2**	**V1**	**V2**	**V1**	**V2**	**V1**	**V2**	**V1**	**V2**	**V1**	**V2**
Mean	2.05	1.64	2.48	1.81	2.53	2.08	2.53	2.76	3.24	3.02	3.83	3.34
Variance	0.07	0.02	0.01	0.12	0.00	0.15	0.00	0.12	0.01	0.04	0.00	0.12
Pearson Correlation	-0.33		-0.38		-0.03		-0.62		-0.52		-0.04	
t Stat	2.60		3.68		2.60		-1.30		1.94		3.06	
P(T¡=t) two-tail	0.06		0.02		0.06		0.26		0.12		0.04	
Professional and Web camera												
Client			**Sharp 2.0**		**Sharp 4.0**		**Sharp 8.0**		**Sharp 10.0**		**Sharp 20.0**	
	V1	**V2**	**V1**	**V2**	**V1**	**V2**	**V1**	**V2**	**V1**	**V2**	**V1**	**V2**
Mean	1.64	1.96	1.81	2.16	2.08	2.17	2.76	1.99	3.02	1.86	3.34	2.01
Variance	0.02	0.01	0.12	0.01	0.15	0.03	0.12	0.03	0.04	0.07	0.12	0.05
Pearson Correlation	0.03		-0.32		0.01		-0.41		0.40		0.30	
t Stat	-3.82		-2.02		-0.49		3.81		9.99		8.33	
P(T¡=t) two-tail	0.02		0.11		0.65		0.02		0.00		0.00	
Mobile and web camera												
Client			**Sharp 2.0**		**Sharp 4.0**		**Sharp 8.0**		**Sharp 10.0**		**Sharp 20.0**	
	V1	**V2**	**V1**	**V2**	**V1**	**V2**	**V1**	**V2**	**V1**	**V2**	**V1**	**V2**
Mean	2.05	1.96	2.48	2.16	2.53	2.17	3.00	1.99	3.24	1.86	3.83	2.01
Variance	0.07	0.01	0.01	0.01	0.00	0.03	0.02	0.03	0.01	0.07	0.00	0.05
Pearson Correlation	-0.91		-0.36		-0.29		0.81		0.19		0.43	
t Stat	0.61		4.41		4.14		21.87		11.50		19.43	
P(T¡=t) two-tail	0.58		0.01		0.01		0.00		0.00		0.00	

The variant of the *t-test* we used allows the variances of the normal distributions to be different:

$$t = \frac{\overline{x}_1 - \overline{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \tag{6.3.1}$$

where x_1 and x_2 are the means of the populations, that is, the means of the absolute values of the coefficients of the 2D Fourier transforms of the Difference of Gaussians

from the pilot images taken by different types of cameras. The test here cover a set of 30 images from 1 person taken by 3 different camera types. n_1 and n_2 are the sizes of the groups, while s_1^2, s_2^2 are the variances of the samples calculated using the formula:

$$s_1^2 = \frac{\sum (x - \overline{x}_1)^2}{n_1 - 1} \text{ and } s_2^2 = \frac{\sum (x - \overline{x}_2)^2}{n_2 - 1} \tag{6.3.2}$$

The *t-test* assesses the difference between the means of two groups, where a smaller $p - value$ means that the difference between two groups is more significant. In our case, we use the $p = 0.05$ (5%) as a significance threshold. Table 6.3 clearly shows that the DFT spectrum of imposters taken by professional and mobile cameras in most cases is not statistically significant i.e., ($p > 0.05$), meaning that we can reject the null hypothesis. Meanwhile, the p values indicate significant differences between the webcamera on the one side and mobile and professional cameras on the other. As a result, we used the professional camera to construct the database since the quality of its images is comparable to the mobile phone camera and it offers more control of the image photo-shooting process. Nevertheless, in the photo shooting sessions we also captured client images with the mobile phone camera and the web camera for possible later use.

6.4 DURHAM FACE Database

The DURHAM FACE Database contains face images from 21 people. All photo-shooting sessions took place in the Imaging Laboratory of Durham University and for each participant a total of 50 photos were taken using a professional Canon EOS Rebel T3i (600D) with a 18-250mm lens.

Client images: The camera was mounted on a tripod and operated with the default auto-focus settings at $5,184 \times 3,456$ resolution. To isolate as much as possible the effect of the image sharpening that we applied to create the processed imposter images, all client images were taken in a frontal view, with neutral expressions, under uniform illumination and background conditions. The raw client images were cropped down to size 640×640, which gives a good balance between image quality and the speed of training and testing the classifier. For our purposes, it was im-

portant to avoid any resizing of the images, since that would mean an extra image processing operation with a largely unpredictable and difficult to account for effect. To achieve this, all participants were seated between 1m and 1.25m away from the camera, at which distance it was possible to obtain tightly cropped face images of the required 640×640 resolution.

Imposter images: The imposter images were created from client images as shown in Figure 6.1 (B). For each subject, an arbitrarily chosen client image, which later would not be used for either training or testing, was printed on A4 paper using a Ricoh 4500 Photocopier. The printed paper was pinned on a board and a series of photos were taken with the camera's auto-focus mechanism re-enabled between any two shots.

Figure 6.4 shows imposter images produced from small size printed images re-captured from close distance by the camera of an iPhone 6s with non-blocking auto-focus mechanism. When the auto-focus mechanism fails the imposter image becomes extremely blurry, see Figure 6.4, demonstrating the importance of including the type of camera focus mechanism in the design protocol of the database, for example blocking auto-focus, non-blocking auto-focus, manual-focus or fixed-focus.

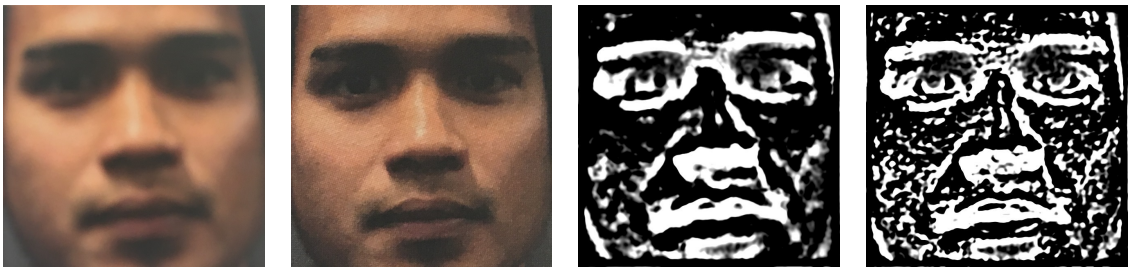


Figure 6.4: Imposter images captured with an iPhone 6s camera from a distance of 6cm (i) and 9cm (ii). (iii)-(iv) DoG for the images (i)-(ii) with the $\sigma_1 = 4$ and $\sigma_2 = 8$.

Processed imposter images: The processed imposter images were created from the client images as shown in Figure 6.1 (B). The same arbitrarily chosen client image used to create the imposter images was sharpened using Matlab's *imsharpen* function with parameter value $\sigma = 8.0$. Then, it was printed on the Ricoh 4500 Photocopier and the same procedure that created the imposter images was followed.

The components of the resulted database of 21 individuals are: 1,050 client images, 630 print on paper and recapture imposter images and 630 more imposter images processed with $\sigma = 8.0$ amount of sharpening. This original database was later extended with the capture of another 30,765 images displayed on screen, as described in Section 6.6.

Figure 6.5: Sample from the DF photo-shooting sessions. **Left:** Creating client images. **Right:** Creating imposter images.

Figure 6.5 shows instances of the photo-shooting session. The first column of Figure 6.6 shows client images from the DURHAM FACE Database. The sharpened client images in the second column, which are not part of the database, exhibit higher contrast and some sharpening artifacts. The third column shows imposter images from the database; they are more blurry than the client images. Finally, the fourth column shows sharpened imposter images from the database, which show the highest visual similarity with the client images. Indeed, the direct digital sharpening with Matlab's *imsharpen* followed by a procedural blurring by printing them on a paper and recapturing them, seems to be a great extent to cancel each other.

6.5 Testing

We tested the DURHAM FACE Database by running on it the well-known liveness test proposed in [177], which is conceptually simple and easy to implement. While testing with various liveness tests would have given us a better understanding of the behaviour of the database, we note that the fundamental nature of the mathematical

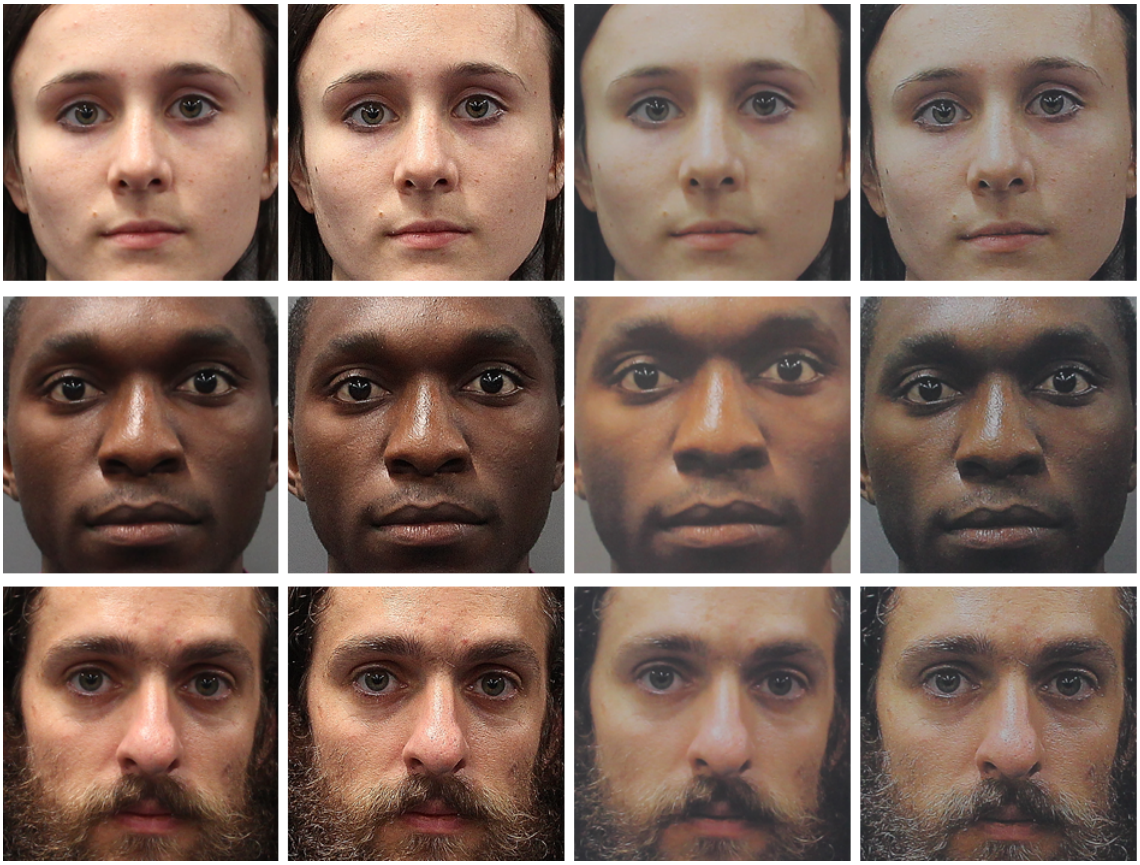


Figure 6.6: Samples from the DURHAM FACE Database for three different subjects. **Left to right:** a client image; a sharpened client image; an imposter (recaptured) image of the client image; an imposter of the sharpened client.

and statistical tools employed by that test, namely differences of Gaussians of images and sparse logistic regression, make it a suitable choice as a representative liveness test. The detailed description of the test can be found in Section 5.2.1.

In all tests, the standard deviations of the differences of Gaussians were set at $\sigma_1 = 4$ and $\sigma_2 = 8$. The results are shown in Figure 6.7. Each diagram consists of two ROC curves, one showing the performance of the classifier in distinguishing between client and imposter images and the other in distinguishing between client and sharpened imposter images. The four diagrams correspond to four different designs of the training set which may contain:

(i) client and imposter images from all 21 subjects,

(ii) client and sharpened imposter images from all 21 subjects,

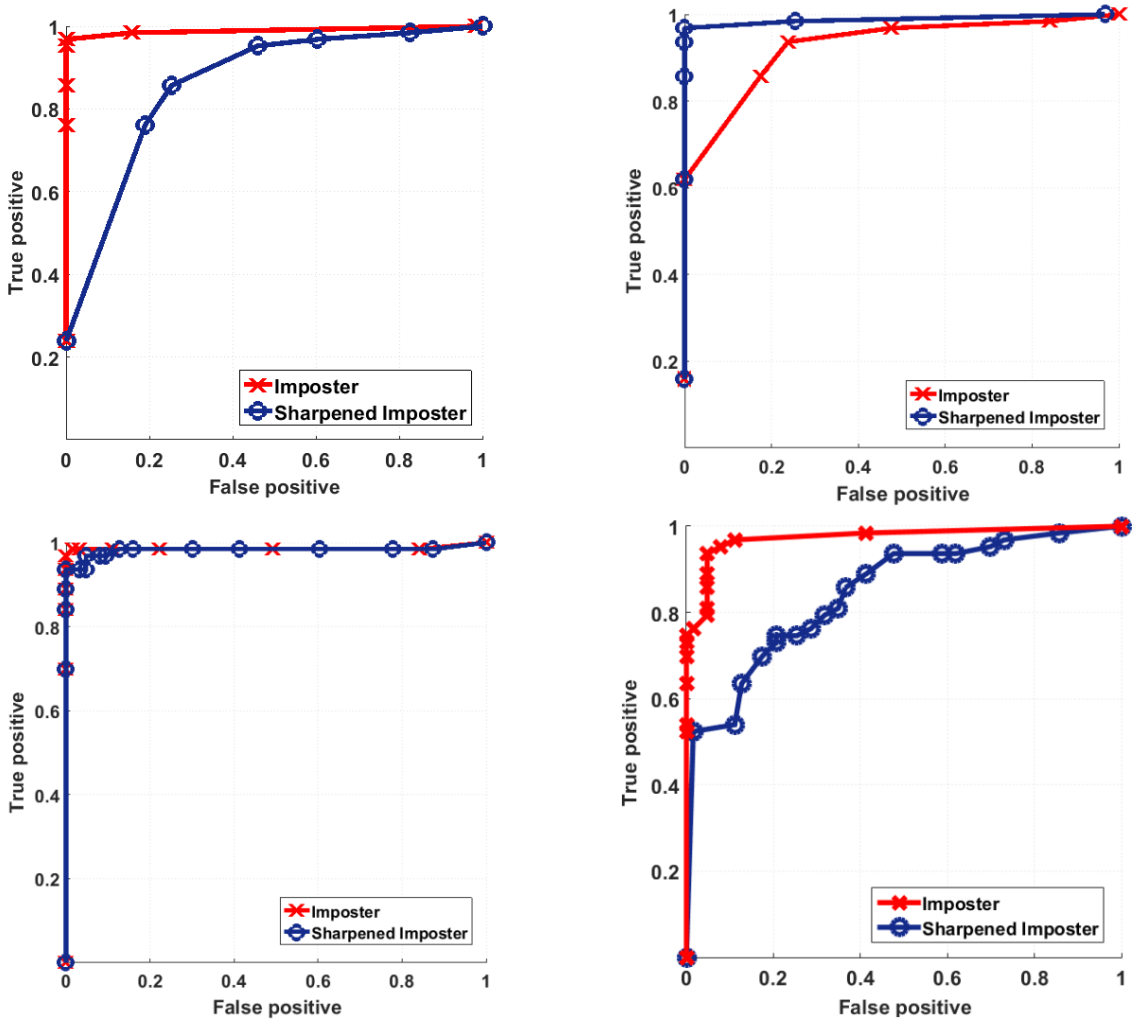


Figure 6.7: In each diagram the two ROC curves show the performance of the algorithm in distinguishing between client images and imposter or sharpened imposter images, respectively. **(i)-(iii)** The training set consists of client images and: (i) **top-left:** imposter images, (ii) **top-right:** sharpened imposter images, (iiii) **bottom-left** both imposter and sharpened imposter images, from all 21 subjects. **(iv) bottom-right** The training set consists of client and imposter images of 15 subjects, while images from the other 6 subjects are used for testing.

(iii) client, imposter and sharpened imposter images from all 21 subjects,

(iv) client and imposter images from 15 of the subjects, while the test set contains images from the other 6 subjects.

Since client images have always to be included in the training set, Figures 6.7(i)-(iii) cover all three cases regarding the content of the training set: imposter images, sharpened imposter images, or both. Figure 6.7(i) verifies our main hypothesis, showing that the performance of the liveness test decreases considerably when the attacker uses sharpened imposter images. Indeed, the large gap between the two curves indicates a significant drop in the performance of the liveness test, which is largely due to the fact that the classifier was trained to distinguish between client and imposter images and not between client and sharpened imposter images. This can be seen in Figure 6.7(ii), where the classifier is trained to distinguish between client and sharpened imposter images and as a result the liveness test is much more efficient against attacks with such images. In Figure 6.7(iii), the classifier is trained to distinguish between client images and imposters of both types and we notice that its discriminative ability decreases slightly only against attack with sharpened imposter images. This was again an expected result, since the level of similarity between sharpened imposter rather than common imposter images, and client images is higher than the level of similarity between client and imposter images, as it is clear in Figure 6.6, distinguishing between sharpened imposter and client images is a harder task when no task, is given preferential treatment during training.

From the results in Figures 6.7(i)-(iii) we conclude that by sharpening the client image before printing and recapturing it to create imposter images, attackers can significantly increase their chances of evading the Tan et al. liveness test. On the other hand, a very simple and largely effective countermeasure is to train the classifier not only with imposter but with sharpened imposter images too. In that case, there is only a slight decrease in the performance of the classifier on sharpened rather than common imposters, which can be explained by the higher similarity between client and sharpened imposter images, which makes the distinction between these two classes an intrinsically more difficult task.

Figure 6.7(iv) shows the results when the classifier is trained with a cross-subject independent set of client and imposter images. That is, the subjects are partitioned into two non-overlapping subsets and images from the first subset are used for training while images from the second subset are used for testing. We notice that the cross-subject independence is a strong assumption which is not used in the relevant literature since liveness tests usually run in parallel to face recognition systems and a positive classification of a subject by a face recognition system implies the presence of their images in the database. Nevertheless, Figure 6.7(iv) shows that even with the assumption of cross-subject independence, our claim that imposter attacks with sharpened images are more effective than common imposter attacks is valid.

6.5.1 Printer Image Processing

Apart from pointing out that the strength of malicious presentation attacks can be underestimated if image processing operations before image printing are not considered, the above results are also relevant in the setting of the common imposter attacks via the advanced and largely automatic image processing functionality of modern printers and cameras. As most printers and cameras make use of proprietary image processing technology, the study of the effect of printer and printer settings on the performance of the liveness test is very challenging.

Figure 6.8 illustrates the effect of printer choice through an example. Indeed, visual inspection of the printouts of the same digital image by two different printers reveals significant differences and as a result the differences of Gaussians of the two images are also significantly different. From each of the two printouts we created 10 imposter images and computed the probabilities of the image to be imposter, using the same classifier as in Figure 6.8(i). When the Ricoh 4500 printout was used, that is, the printer that created the printouts for the imposter images of the training set, the average probability was 0.929. On the other hand, when Bizhub c654e printer was used the average probability dropped sharply to 0.083. Thus, the example indicates that the choice of printer and printer settings can affect the performance of the liveness test and perhaps, in agreement with what we found in our main experiment, using a variety of printers and printer settings to create the

training set, can increase the generality of the classifier.

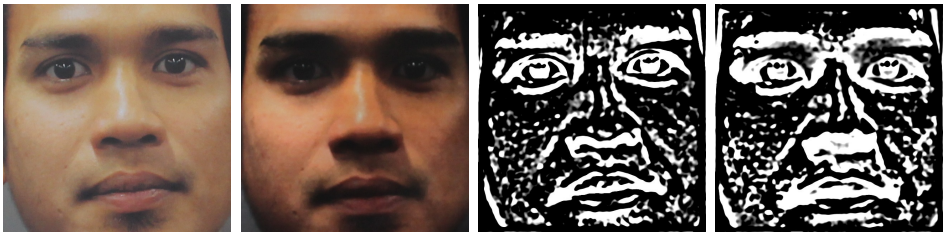


Figure 6.8: (i)-(ii) imposter images from printouts by Ricoh 4500 and Bizhub c654e, respectively. (iii)-(iv) the corresponding differences of Gaussians for $\sigma = 4$ and $\sigma_2 = 8$.

6.6 Database Extension

The DURHAM FACE database was expanded to include images of various imposters displayed on digital screens, besides the real client images and images of imposters and sharpened imposters printed on a piece of A4 paper of the original database. Imposters were created by playing in front of the camera a video clip at a frame rate of 60 fps, showing sharpened imposter images at various amounts of sharpening: 0.0, 2.0, 4.0, 8.0 and 16.0. We used a Toshiba Ultra laptop's digital screen, under uniform lightning conditions, with the brightness of the display set to maximum. The same professional camera, the Canon EOS Rebel T3i (600D) with a 18-250mm lens, was used, mounted to a tripod, to capture the video clip. Figure 6.9 presents samples of imposters on the digital display, with different amounts of sharpening. Our extended database contains a total of 30,765 imposter images recaptured from clients shown on a digital display, all at a resolution of 320 x 320. We chose 5 random clients from each of the 21 subjects in the database, and then, the images were sharpened before being played in-front of the professional camera as an MS Power Point slide show, with a fixed 2-second automatic timer for slide moving. As a result, 300 frames for each individual were produced before some frames around each slide transition were removed. Hence, approximately a total of 293 frames of each type of processed imposters were produced for each subject. The images were cropped and classified accordingly.

This extension to the DURHAM FACE database, was used in Chapter 8, where the need for a larger dataset became clearer, as we wanted to evaluate an approach based on a pre-trained Convolutional Neural Network against various processed imposter image attacks.

6.7 Good Practices in Database Design

To optimise the design of a facial spoofing database to be used in the evaluation of liveness tests, we recommend good practices.

1. **The camera:** there are hundreds or thousands of types of digital cameras, each one with a different set of features that might lead to significant differences in the properties of the captured images. Hence, considering as many camera types as possible is one of the keys towards a good database design.

2. **Focus mechanism:** cameras can operate either on manual-focus, fixed-focus, or auto-focus. In the manual-focus, the photographer is responsible for setting up the focus mechanism's options, in which case their expertise plays major role in achieving the best settings. Fixed-focus is the least used mechanism in high end cameras, but still used in webcameras. In this case we expect to have imposter images with less sharp features if we use a smaller than natural size printed image and bring near to the camera to compensate. That could seriously affect the quality of the database as imposter images would simple be blurry out of focus images. Auto-focus is the most widely used option in most modern cameras. For some various reasons, the auto-focus mechanism might encounter problems and focus could go wrong. This might create a few problematic imposter images within the database.

3. **Camera hold:** the camera can be either in a fixed position or hand-held. Most images available on social media are taken by devices being held rather than, for example, being mounted to tripods. As a result, these images might often be slightly blurry due to the mini-scale camera shakes during photo-shooting.

Figure 6.9: Samples from the DURHAM FACE Database of imposters from a digital display, of five different subjects and with different sharpening values. **Left to right:** different sharpening amounts have been applied to imposter images; 0.0, 2.0, 4.0, 8.0, and 16.0.

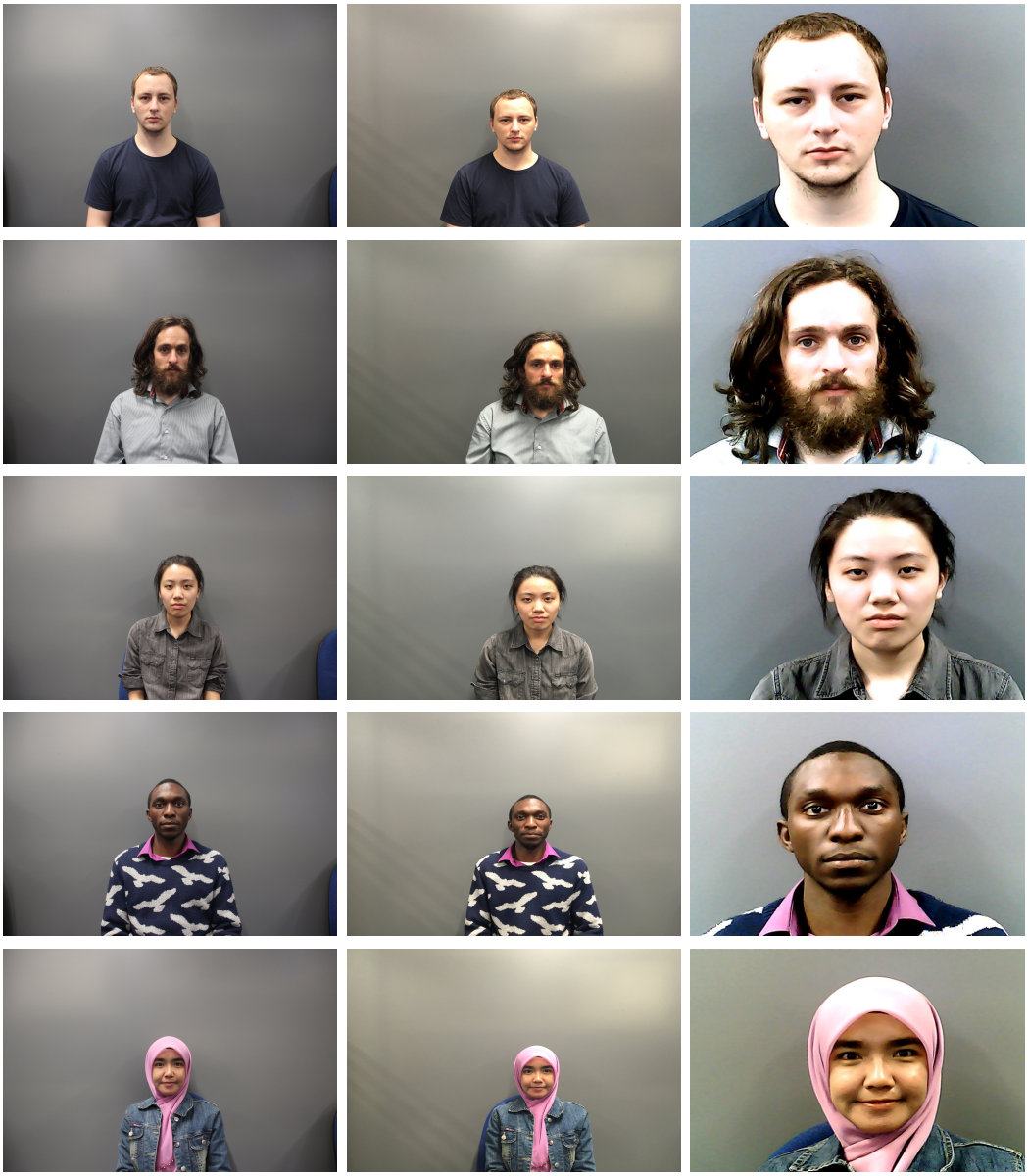


Figure 6.10: Samples from the original photo sessions for the DURHAM FACE Database of five different subjects taken with different cameras. **Left to right:** professional camera, mobile phone and webcamera.

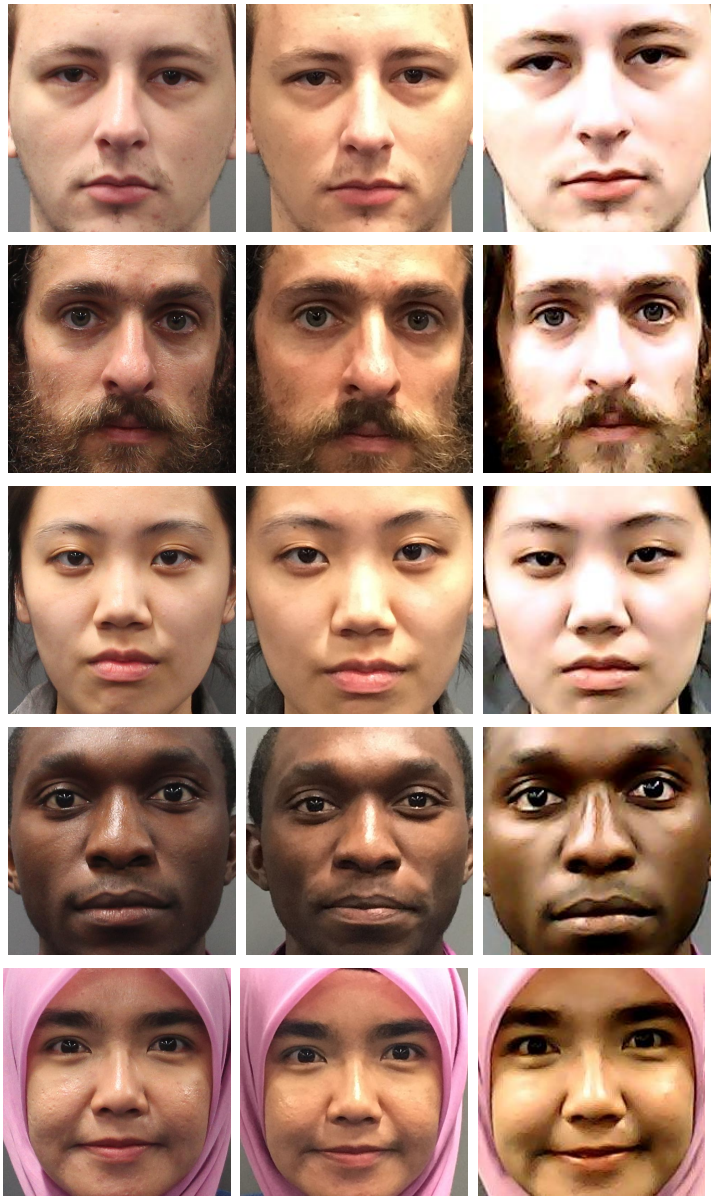


Figure 6.11: Samples from the DURHAM FACE Database of five different subjects taken with different cameras, after cropping. **Left to right:** professional camera, mobile phone and webcamera.

4. **Object size and distance:** from the camera producing an imposter image from a real client image is a complex task, requiring printing the client images on a paper or displaying them on a digital display and setting them up for a photo-shooting session after finding an appropriate for the size of the image distance from the camera. Having objects too faraway or very close to the camera will result in imposters of poor quality. Therefore, a detailed study of appropriate image size and distance from the camera must be undergone prior to the creation of the database.

5. **Printer ink and paper:** printed images may vary duo to ink quality variations, while the type, colour and the quality of the paper also plays a significant role in the appearance of the printed image.

6. **Digital display brightness:** a potentially important factor when presenting an imposter on a digital screen, as incorrect brightness levels can lead to imposter images being altered in undesired ways.

6.8 Conclusions

In this chapter we presented the DURHAM FACE Database, a new publicly available face image database for testing anti-spoofing algorithms. The main novelty in its design is the inclusion of processed imposter images, justified under the assumption that attackers can easily process the client images before printing them for presentation attacks, and in particular, they may sharpen them in order to counteract the blurring induced by the printing and recapturing process. Our tests show a significant decrease in the performance of a standard liveness when sharpened imposter images are used, however, most of the performance loss can be restored by the simple measure of including sharpened imposter images in the training set.

In the future we plan to extend the database with more types of processed imposter images. In particular, instead of processing the client images with existing, simple or sophisticated, image processing operations, we would like to reverse engineer the process and thus, be able to directly compute images which under printing

and recapturing produce imposter images that are as close as possible to the the original client images.

Chapter 7

Signal-Noise Analysis of Face Anti-spoofing Algorithms

In this chapter we study the performance of two face anti spoofing algorithms; the Tan et al. and a tailor made ANN based algorithm, under the assumption that the attacker applies digital image processing operations on a face image before using it for the spoofing attack. We treat the main variable of the image sharpening algorithm as parameter and capture the behaviour of the algorithm's performance in terms of this single parameter by a statistical model of its signal noise decomposition, rather than by computing ROC curves. In particular, we propose beta distribution models for both the signal and the noise and show how the sharpening of the image used for the attack affects the parameters of these beta distributions.

7.1 Introduction

Face recognition is a prime candidate technique in biometric identification applications requiring real-time, reliable and unobtrusive user authentication without the use of specialized hardware. On the other hand, it is also considered vulnerable to spoofing attacks, in the form, for example, of imposters gaining access to the system by holding in front of the camera a stolen image of the user. This vulnerability means that user authentication through face recognition is still mostly confined to applications with low security requirements, such as low security level mobile phone

log-in, or applications in highly controlled environments, such as airports.

Liveness tests are countermeasures to spoofing attacks. One limitation shared by the majority of the current approaches to the development of liveness tests is the tacit assumption that the imposter will use the stolen image or video as it is, i.e., without previously processing it in order to increase the effectiveness of the attack. Moreover, this tacit assumption is carried over from the development to the evaluation of liveness tests. Thus, the most popular image and video databases for evaluating liveness tests, such as the NUAA [177], PRINT-ATTACK [14], REPLAY-ATTACK [33] and CASIA [208] consist of imposter images or videos captured from unprocessed photographs or videos of the users. The DURHAM FACE database [131] contains processed imposter images, but it is considerably smaller than the previous ones.

The processed image imposter attack was proposed in [132] and tested with direct sharpening of imposter images from the NUAA database. Our testing approach was further validated in [131] with the creation of the DURHAM FACE database containing sharpened imposter images produced as photographs of printed sharpened client images. Here, we use for our evaluation two liveness tests, a classic one based on a Sparse Logistic Regression (SLR) classifier applied on a differences of Gaussians (DoG) feature space, and one that is tailored-made for the purposes of this chapter, based on Artificial Neural Networks (ANNs) applied on raw images.

SLR was chosen as a mathematically well understood technique with predictable behaviour. It is considered as a baseline machine learning technique which performs robustly in this type of problems. Moreover, Lambert's cosine Law can be used to justify the ability of DoG in distinguishing between the light reflections from real faces and recaptured images [177]. On the other hand, ANN was chosen as a generic machine learning algorithm, without any distinct feature extraction step based on assumptions about the optics of the problem. The use of convolutional neural networks as in [201] and [107] was also considered. However, this promising research direction seems at the moment to be hampered by a lack of face anti-spoofing specific data for CNN training. And thus, we first opted for a simpler ANN based generic algorithm which was sufficient for our purposes.

We aim at a better understanding of the behaviour of liveness tests against processed imposter image attacks, employing a more detailed statistical modeling of the output of the classifiers, beyond the plotting of empirical ROC curves. In particular, for each instance of the attack, corresponding to a specific value of the main parameter of the image sharpening function, we model the output of the classifier on the client and the imposter images with two *beta distributions*, and study the relation between the sharpening parameter and the parameters of the beta distribution. For example, if we train the SLR classifier with target values of 0 for imposters and 1 for *client* images, then, by increasing the amount of sharpening of the imposter images, the imposter beta distribution sometimes becomes bimodal, i.e., the middle values of the classifier's output on imposter images are pushed towards the ends of the support [0,1]. The direct consequence for an attacker aiming at maximizing the effectiveness of their attack is that against a strict system operating at the left hand end of the ROC curve, sharpened imposter images will be more effective, while against loose systems operating at the right hand end of the ROC curve will be less effective.

Perhaps, the most appropriate framework for presenting our approach is the one provided by the *signal detection theory* [118], with the classifier's output on clients being the *signal* and its output on imposters being the *noise*. From the signal detection point of view, the sharpening of the imposter images decreases the mean and increases the variance of the distribution of the noise, producing two effects that act in opposite directions. On the one hand, the decrease in their mean value moves imposters further apart from clients and thus, makes them easier to detect, while the increased variance blends them more with the clients, making them more difficult to detect.

Contributions: The main contributions of the chapter are: (i) The use of a signal-noise decomposition model based on beta distributions for studying the behavior of liveness tests under processed image attacks with the amount of sharpening treated as a parameter. (ii) An evaluation of this model, both with a classic liveness test and with a tailor-made one, aiming at exhibiting the variety encountered in the behaviour of different liveness tests under simple image processing operations.

Limitation: The main limitation of the chapter is that following [132] we evaluate our approach indirectly, that is by direct processing of the imposter images, instead of processing client images, printing them and taking photos of them, which will then be used to produce imposter images. However, we note that the validity of this indirect approach has already been verified in [131]. Moreover, it is based on the very reasonable assumption that if a digital image is sharper than another, then it will most likely remain the sharper one after both images are printed on paper and recaptured on a camera.

The rest of the chapter is organized as follows. In Section 7.2 we describe the two liveness tests we use and the methodology for analyzing their effectiveness under processed image attacks. In Section 7.3 we present and discuss the results and we briefly conclude in Section 7.4.

7.2 Method

The imposter images are processed with Matlab's *imsharpen* function, using various values of the function's main parameter which controls the amount of sharpening. Next, each set of sharpened imposter images, together with the set of client images is used to test two different anti-spoofing algorithms. Finally, the outputs of these algorithms are analyzed with a signal-noise decomposition method, that is, by fitting two beta distributions on the outputs of the classifier, one on the imposters and one on the clients, respectively.

Notice that *imsharpen* retains visual facial information well and we do not expect problems with the face recognition part of the system, even for large values of the main parameter, see Figure 7.1

7.2.1 Anti-spoofing Algorithms

The first classifier we use is based on sparse logistic regression (SLR) for the machine learning part, in conjunction with differences of Gaussians for feature extraction. It was proposed in Tan et al. [177] and became a popular choice for evaluating algorithms and databases [116, 208], not only because it is easy to implement, but

Figure 7.1: **Left to right:** Client, imposter and imposter sharpened by 1.0, 5.0 and 50.0, respectively.

also because the employed techniques, i.e., logistic regression and Gaussian smoothing, are mathematically well understood and as a result have, generally, predictable behaviour.

Here, we set the standard deviations for the difference of Gaussians to $\sigma_1 = 0.5$ and $\sigma_2 = 1.0$ and use a value of $\lambda = 0.05$ as our regularization constant. For an input test image x, the output of the trained sparse logistic regression model is:

$$\frac{1}{1 + \exp(w^T x + b)} \tag{7.2.1}$$

where w is the weight vector, and b the intercept value. The output can be interpreted as the posterior probability of the image x to be a *client*.

The second anti-spoofing algorithm we use is an Artificial Neural Network (ANN), trained on the raw images of the database. To the best of our knowledge, shallow ANNs have not been used before in this specific setting but, nevertheless, in our experiments they performed better than SLR. Given the high dimensionality of the data compared to the size of the database, we opted for a very simple design consisting of a single hidden layer with 10 nodes, the number of which was chosen after extensive trial and error experimentation. The hidden layer transfer functions were tan-sigmoids and the output transfer function log-sigmoids, and thus, the output values were again in the range [0,1]. For training our dataset $\mathbb{D}$, we use one of the most popular simple in terms of theoretical analysis algorithms, the scaled conjugate gradient-descent (SGD). Specifically, we used the TRAINSGD from Matlab's *nntool* with the default parameters of $\lambda = 10^{-7}$ and $\sigma = 10^{-5}$. The loss function was:

$$loss(\mathbb{D}) \triangleq \frac{1}{N} \sum_{i=1}^{N} l(\mathbf{X}^{(i)}, y^{(i)}) \tag{7.2.2}$$

where,

$$l(\mathbf{X}^{(i)}, y^{(i)}) = |y^{(i)} - \hat{y}^{(i)}|^2 \tag{7.2.3}$$

where $y^{(i)} \in \{0, 1\}$ is the label of sample $\mathrm{X}^{(i)}$ and $\hat{y}^{(i)}$ is the value returned by the ANN on input $\mathrm{X}^{(i)}$.

The stopping criteria were again the Matlab's default, that is, terminate when the magnitude of the gradient becomes smaller than $= 10^{-6}$, or 6 consecutive validation checks show no improvement. Typically, the algorithm was terminating after about 50 epochs. In testing the two classifiers we applied both *cross-subject* and *within-subject* training. The cross-subject training set consisted of 3,118 clients and 5,293 imposters from 9 subjects, while the test set consisted of 1,367 clients and an equal number of imposters from 6 subjects. The within-subject training and test sets each consisted of 1,000 client images and an equal number of imposters from 15 subjects. In all cases we used the 64×64 grayscale images of the NUAA database.

In all cases the input of the ANNs were the raw image data. We experimented with various extracted image features, such as differences of Gaussians for inputs, but there were no improvements in the results.

7.2.2 Signal-Noise Decomposition

In the face anti-spoofing literature, the most common way to evaluate the performance of classifiers is by plotting the empirical ROC curves. In our case, ROC curves corresponding to different amounts of sharpening generally intersect. In fact, in some cases the whole family of ROC curves seems to pass from a single point. While there is nothing unusual in that behavior, it suggests that we might get further insights into the performance of the classifiers by studying the signal noise decomposition of their outputs.

Following standard signal detection terminology, we call the output of the classifier on imposter images *noise*, and the output on client images *signal*. The classifier's output on the imposter test set is a sample of the noise and it is modeled with a beta distribution $Beta(\alpha_n, \beta_s)$. Similarly, the classifier's output on the client test set is a sample of the signal, modeled with another beta distribution $Beta(\alpha_s, \beta_s)$. Here,

the parameters of the beta distribution are computed with the maximum likelihood method, using Matlab's *betafit* function.

We note that the distributions of the noise and the signal contain more information than the ROC curve alone. Indeed, from the distributions $P_n(t)$ and $P_s(t)$ of the signal and the noise, which in our case both have support [0,1], we can obtain all the points of the ROC curve

$$(x(t_0), y(t_0)), \quad 0 \leq t_0 \leq 1$$

as,

$$x(t_0) = \int_{t_0}^{1} P_n(t)dt \qquad y(t_0) = \int_{t_0}^{1} P_s(t)dt \tag{7.2.4}$$

while, generally, from the ROC curve we can not retrieve the two distributions.

We also note that the use of the beta distribution for modeling the output of binary classifiers is very common in the literature. For example, [158] uses beta distributions to model the performance of a face recognition algorithm under repeated trials on the same user. In the relevant literature, the reasons most often cited for the choice of a beta distribution are, firstly, that its support is, conveniently, the interval [0,1], and secondly, that it is the conjugate prior of the binomial distribution, i.e., the distribution expected from a Bernoulli trial.

A simpler alternative to the modeling of the signal and noise samples with beta distributions would have been to study the behavior of a distance function between the two samples, in which case, distance minimization could then be used by the attacker as a strategy for choosing an optimal amount of sharpening. However, regardless of the choice of distance function, this simpler approach fails to describe the effect of sharpening as a trade-off between the maximization of true positives and the minimization of false positives.

Table 7.1 shows the Hellinger distance

$$(1 - \sum_{i=1}^{20} (h_{im}(i) \cdot h_{cl}(i))^{1/2})^{1/2} \tag{7.2.5}$$

between the twenty-bin histograms of the classifiers'outputs. We notice that apart from cross-subject SLR, in all other cases the distance function indicates that small

Table 7.1: Hellinger distances between the twenty-bin histograms of the client and imposter outputs.

	sharp 0	sharp 1	sharp 5	sharp 50
SLR cross	0.4633	0.3959	0.2763	0.2399
SLR within	0.8242	0.8307	0.8542	0.8303
ANN cross	0.6635	0.6641	0.6482	0.6496
ANN within	0.8929	0.8976	0.8941	0.8093

amounts of sharpening slightly weaken the attack. The reason for this counter-intuitive result is that sharpening makes some of the large output values of these classifiers, which for any reasonable choice of threshold would have anyway been classified as clients, even larger.

7.3 Results

Figure 7.2 shows the ROC curves of the two anti-spoofing algorithms under cross-subject and within-subject training for various amounts of sharpening. In most cases, and especially in the cases of within-subject training, visual inspection is not particularly revealing since all the curves approach quickly the top left of the ROC box. The only exception is the SLR classifier under cross-subject training where it is clear that the sharpened image ROC curves are lower until around a 0.4 value for the false positive rate and then climb higher, see Figure 7.2 (left). That behavior suggests that a sharpened imposter image attack should be used against strict security systems operating at the left end of the ROC curve and be avoided against loose systems operating at the right end of the ROC curve. The shape of the ROC curves in Figure 7.2 (left) also suggests that the use of the area-under-the-curve metric, which has the disadvantage of not making the distinction between the left and the right end of a ROC curve, is not suitable here.

Figure 7.3 shows the twenty-bin histograms of the outputs of the classifiers for client, imposter and sharpened imposter images. We notice that several of the histograms are U-shaped or J-shaped and that the effect of sharpening on the imposter

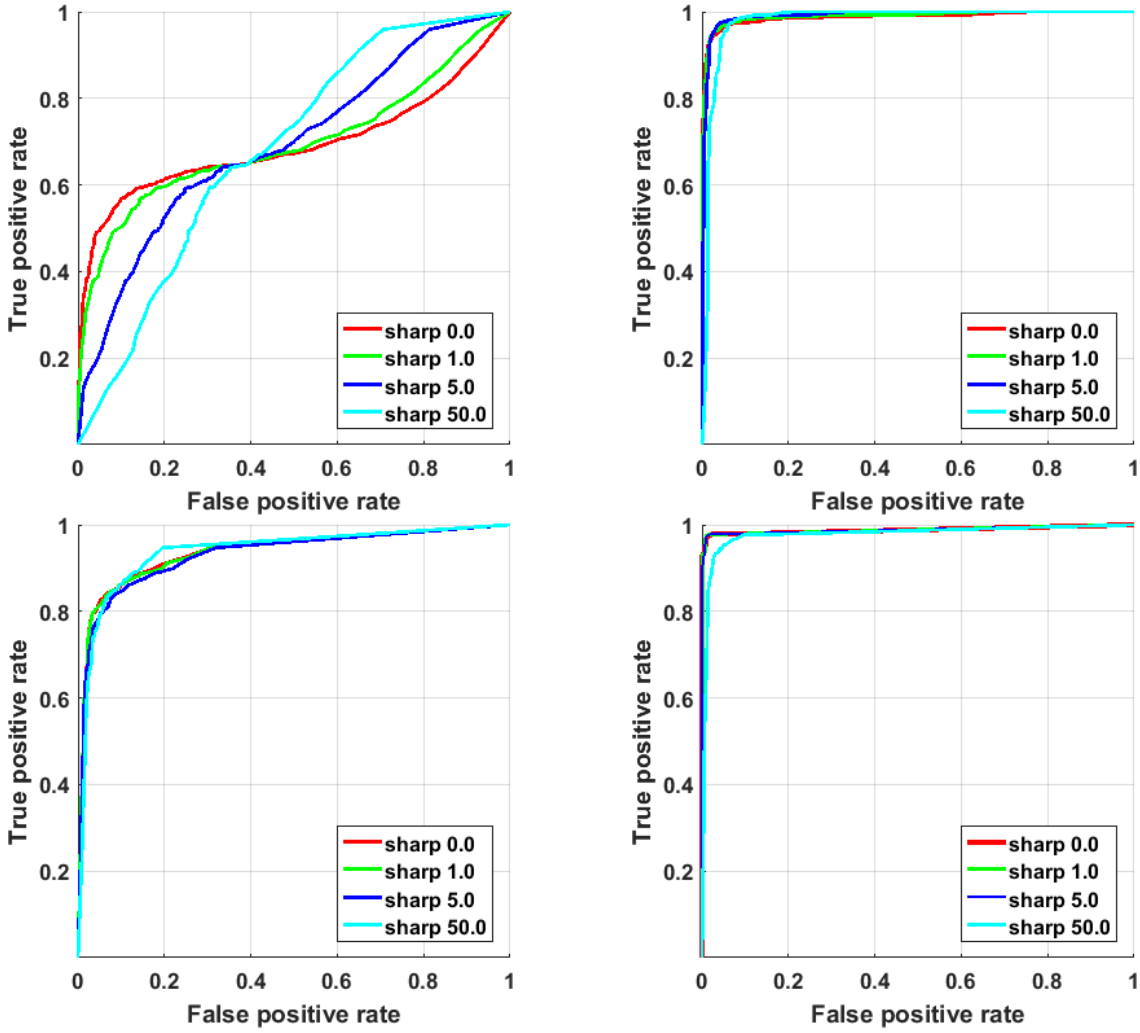


Figure 7.2: **Top-left:** SLR cross-subject, **top-right:** SLR within-subject, **bottom-left:** ANN cross-subject, and **bottom-right:** ANN within-subject.

images can be described in a simple and intuitive way as pushing the output values away from the middle and towards the ends of the range [0,1]. The variation in the shapes of the histograms suggests the use of a two-parameter probability distribution for modelling. Moreover, it shows that the simpler alternative very often used in practice, namely, estimating the empirical mean and variance of the samples, is not suitable here. For a comprehensive discussion of the limitations of ROC curve, see [66]. Indeed, depending on whether the shape of the initial imposter histogram is unimodal or bimodal, pushing the mid-values towards the two ends of the range may increase or decrease the mean and may increase or decrease the variance. Thus, while a beta distribution is completely defined by its mean and variance, the mean

Table 7.2: Maximum likelihood values estimated for α and β.

	SLR cross		SLR within		ANN cross		ANN within	
	α	β	α	β	α	β	α	β
client	0.4704	0.3641	3.2738	0.6726	0.6173	0.2763	1.4747	0.2902
sharp 0	0.7752	1.9158	0.7111	5.0419	0.1879	1.6616	0.2471	17.5246
sharp 1	0.59	1.3654	0.5762	5.3962	0.1852	1.6336	0.2388	17.8584
sharp 5	0.3469	0.7048	0.3099	4.2685	0.1756	1.3805	0.2166	14.2596
sharp 50	0.2207	0.3987	0.1375	1.3993	0.1416	1.1215	0.1719	1.791

and the variance by their own, without any assumption about the distribution, do not provide sufficient information.

Table 7.2 shows the (α, β) values of the beta distributions of the same data as in Figure 7.4. The corresponding beta distribution plots are shown in Figure 7.4. Assuming that $\alpha \neq \beta$, the mode of the beta distribution, for example, whether it is bell-shaped, J-shaped or U-shaped, depends on whether α and β are smaller, equal or greater than 1, see for example [135].

For imposters and sharpened imposters, we notice that α is always less than 1 and decreases with sharpening, while β in most cases is greater than 1, giving J-shaped imposter distributions. The notable exception is the cross-subject SLR for the two larger amounts of sharpening where β is less than 1 and the distribution becomes bimodal, indicating a significant number of false positives even on very strict operating thresholds. In all other cases, the sharpened imposter distributions remain J-shaped with a right hand tail, however, their tail increases with sharpening, bringing them closer to a U-shaped distribution. The larger tail is also noticeable in the histograms in Figure 7.4.

Regarding the client distributions, we notice that the cross-subject client distributions are U-shaped for both classifiers while the within-subject are J-shaped, indicating a qualitatively significant difference in the measured performance between cross-subject and within-subject training, with within-subject corresponding of course to an easier classification problem.

7.4 Conclusions

Recent research has developed face anti-spoofing algorithms of increased robustness, however, coping with the variability of the conditions and the variability of the possible attacks remains a serious challenge. While this variability challenge is reflected in the ways the benchmark databases are designed and the anti-spoofing algorithms are evaluated, still, database and test set variability has a categorical rather than a parametric form. That is, while the databases and the test sets extracted from them consist of several subsets with varying properties, these subsets do not correspond in any systematic way to different values of a parameter. In this chapter, we tested one standard and one tailor made face anti spoofing algorithm in a parametrized setting, the parameter being the amount by which imposter images are sharpened. We discussed the methodological challenges related to parametric testing and we showed that beta distribution modelling of the client and the imposter test sets gives a novel insight into the strengths and weaknesses of the anti-spoofing algorithms.

In the future, we would like to create a parametrized database for testing anti-spoofing algorithms, with the parameter being related either to attack variability, such as the amount of sharpening of the imposter images, or to conditions variability, such as the lighting of the scene.

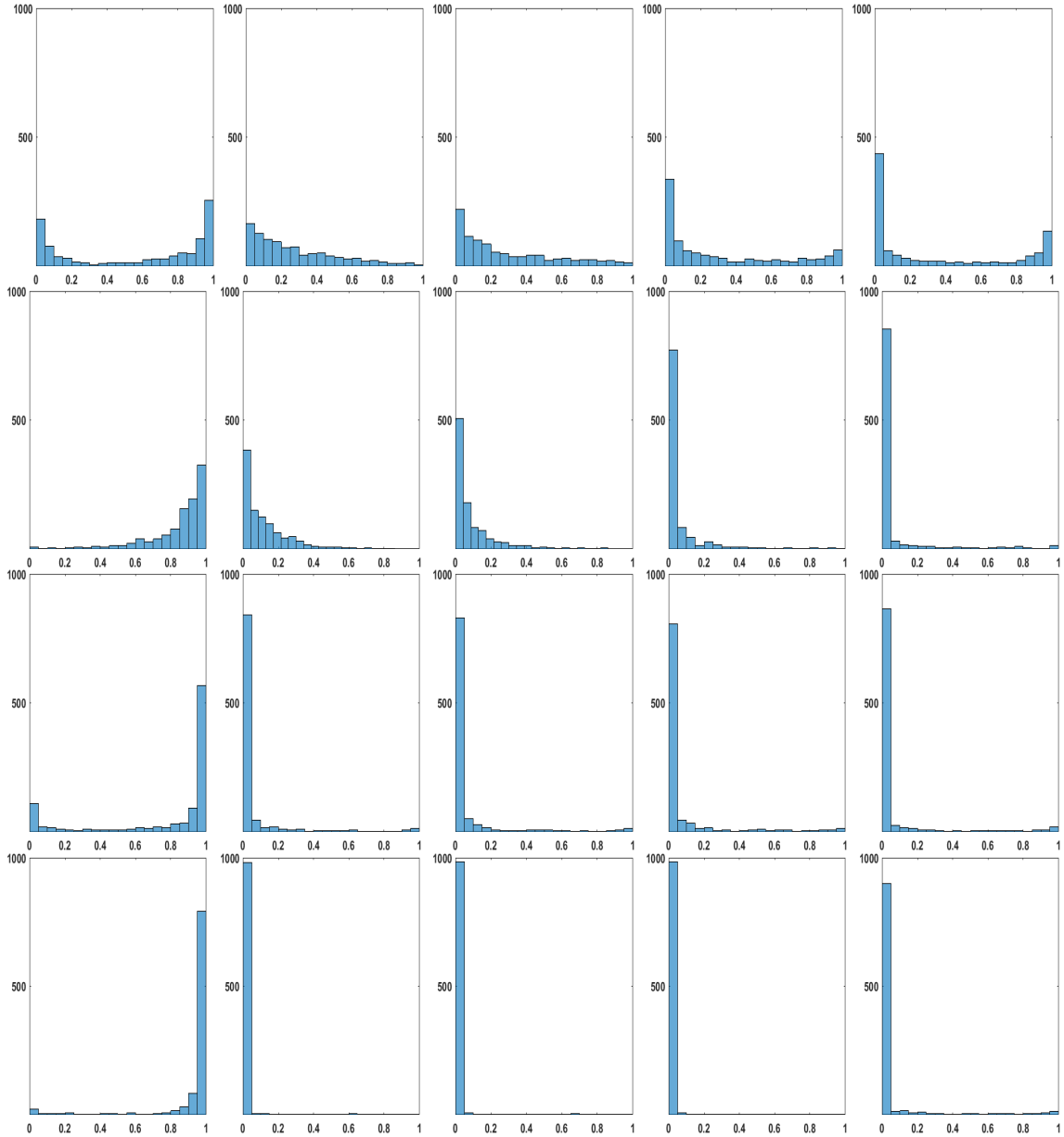


Figure 7.3: **Left to Right:** Twenty-bin histograms of clients, imposters and imposters sharpened by 1.0, 5.0 and 50.0, respectively. **Top to Bottom:** Cross-subject SLR, within-subject SLR, cross-subject ANN and within-subject ANN.

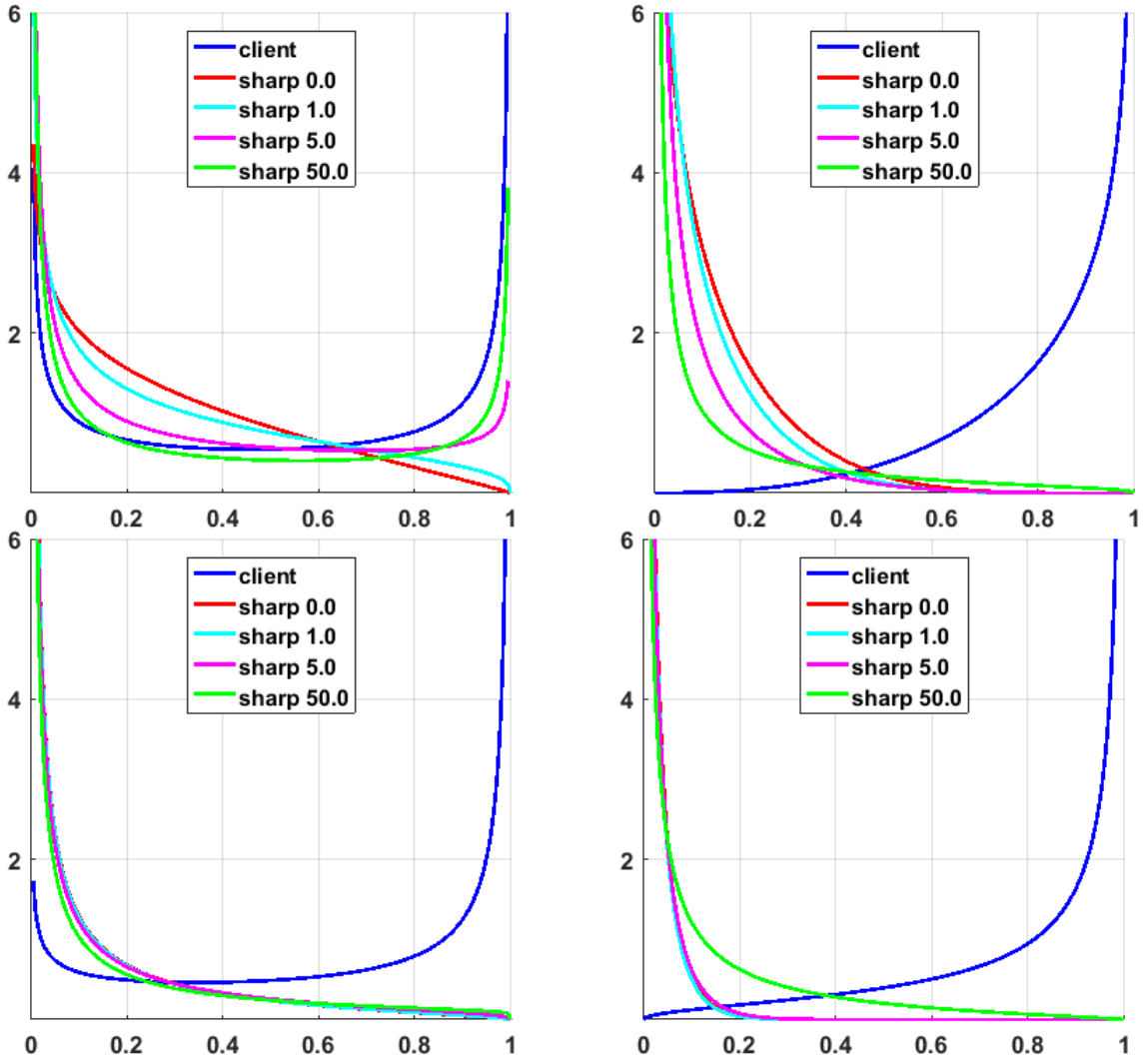


Figure 7.4: Beta distributions fitted to the output values of the trained classifiers. **Top-left:** Cross-subject SLR. **Top-right:** within-subject SLR. **Bottom-left:** cross-subject ANN. **Bottom-right:** within-subject ANN.

Chapter 8

Transfer Learning for Face Liveness Detection

In this chapter we study face liveness detection through a relatively new machine learning technique, that is, transfer learning pre-trained Convolutional Neural Networks. We employ one of the most popular in computer vision applications pre-trained CNNs, the VGG-16, and use it on the client/imposter classification problem of face anti-spoofing. Our aim is to increase the robustness of liveness tests against malicious attacks using imposter images pre-processed by various amounts of sharpening. We evaluate the proposed classifier against our own DURHAM FACE database, described in Chapter 6, which, to the best of our knowledge, is the only one containing sharpened imposter images.

The main contribution of this chapter is a demonstration that a classifier based on the VGG-16 can increase considerably the accuracy of liveness testing, even against attacks with pre-processed imposter images. We also verify that even against liveness tests based on more sophisticated machine learning techniques, such as transfer learning a pre-trained CNNs, the use of sharpened imposter images increases the effectiveness of the attack.

The main limitation of this chapter is the relatively limited scope of the experiment and thus, the preliminary nature of the presented results. We use a relatively small database, CPU rather than GPU computation, and compute accuracy rates rather than ROC curves parameterised by a threshold value.

The rest of the chapter is organized as follows: Section 8.1 presents a brief review of transfer learning and a high level description the architecture of the VGG-16. The classifier and the design of the experiment are described in Section 8.2. The results are presented in Section 8.3 and finally, the conclusions and our intended future work are discussed in Section 8.4.

8.1 Convolutional Neural Networks

The use of Deep Neural Networks (DNNs) have been significantly increased in many domains. These methods are learning methods with multi-levels of representations along the neurons in the network. The Convolutional Neural Network is a class of the DNNs and is normally suitable for analysing visual images.

8.1.1 Transfer Learning

Most machine learning algorithms work under the assumption that the training and test sets are drawn from the same feature space and are two samples of the same distribution. As a result, if for some reason we have to assume that this distribution has changed, then the entire statistical model needs to be updated and the classifier to be rebuilt from scratch, using a new training set. Sometimes this can be impossible, or too costly. For example it can be impossible, or very time consuming to collect new training set, or too computationally expensive to train the model. In practice, especially for high-end machine learning algorithms such as deep CNN, it was been found possible to train a classifier using a training set from one domain and then use it for a classification task in another domain. This technique is known as *Knowledge transfer* or *Transfer learning*, which when used successfully increases the applicability of machine learning, improves its performance and decreases its cost, reducing for example the need for costly manual labelling [140, 206]. In applications, transfer learning is most often used when there are not enough data to train a model from scratch, or when we want to save computational resources at the training stage.

There are two main approaches to transfer learning. In the first approach, we

use the pre-trained model as a feature extractor, with the output of a chosen layer of the pre-trained model being the input of a usually simpler model, which is trained specifically for the new task [149]. The second approach is to fine-tune for the new task the whole of the pre-trained model, or just part of it, using for example backpropagation to adjust the initial weights.

There are few works on face liveness detection using the CNNs. Yang et al. [201] was the first to propose the use of transfer learning based on a CNN for face anti-spoofing. The authors proposed the use of a well-known pre-trained CNN, the AlexNet [99], for feature extraction, and then used an SVM for classification. Their method was tested on the REPLAY-ATTACK database and the in their best case scenario the HTER was 2.81%.

Menotti et al. [125] proposed an approach that uses fine-tuning on the CifarNet pre-trained CNN for various anti-spoofing tasks. Their method resulted on an HTER of 0.76% and 0.00% when used on the REPLAY-ATTACK and 3DMAD databases, respectively.

Lucena et al. [115] proposed a face anti-spoofing approach, called the FastNet. It is based on the architecture of the VGG-16, except for the top layers where authors removed one of the fully connected layers and modified the size of the other two to 256 and 1, respectively. The proposed model resulted on the accuracy rate of 99.04% and an HTER of 1.20% when tested on the REPLAY-ATTACK database, while it achieved a 100% accuracy rate and an HTER of 0.00% when it was tested on the 3DMAD database.

8.1.2 VGG

Today, CNNs are considered the state-of-the-art in solving computer vision problems of various levels of complexity. VGG, also known as OxfordNet, is a very deep neural network trained on ImageNet, which has 15 million high-resolution images collected from the web and labeled by human labelers using Amazons Mechanical Turk crowd-sourcing application. These images are classified into 1000 classes and approximately 22,000 categories. The VGG team secured the first and the second places in the localization and classification tasks of the ImageNet ILSVRC-2014 [153].

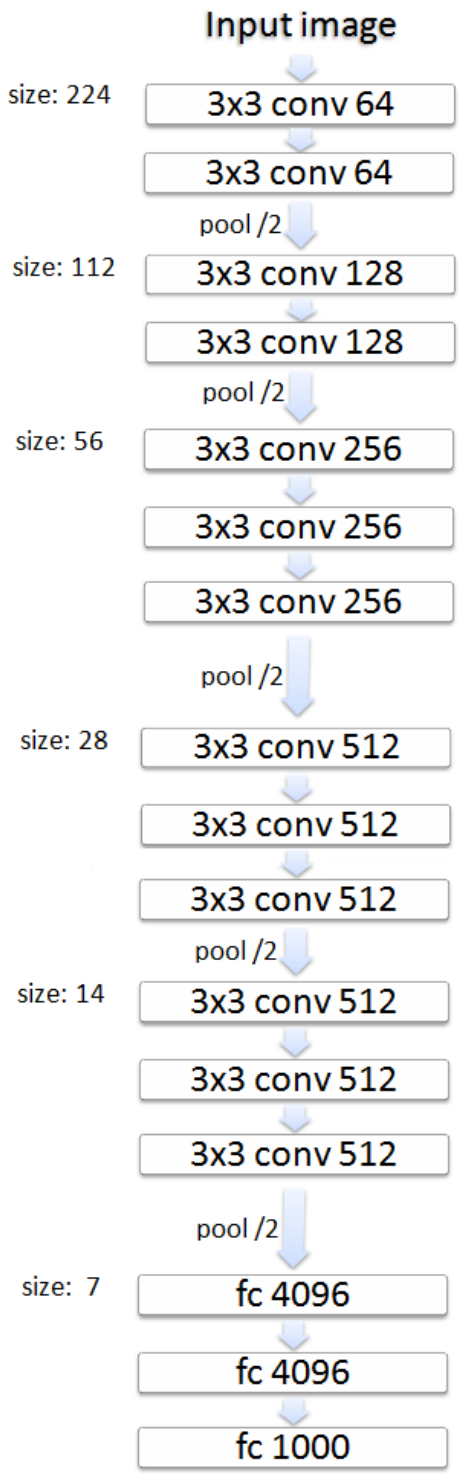


Figure 8.1: The VGG-16 Architecture.

VGG was developed in order to investigate how does the depth of a CNN affects its accuracy in the setting of large scale image recognition. The two standard implementations have a depth of 16 and 19 weight layers, giving the VGG-16 and the VGG-19, respectively [161].

VGG-16 Architecture

As shown in Figure 8.1, the VGG is a 2D CNN with 16 convolutional layers organised in 5 blocks. After multiple filtering operations are performed, the initial $224 \times 224 \times 3$ RGB image is transformed into a $224 \times 224 \times 64$ array in the first block, a $112 \times 112 \times 128$ in the second, $56 \times 56 \times 256$ in the third, $28 \times 28 \times 512$ in the fourth and $14 \times 14 \times 512$ in the fifth. At each convolutional layer, all filtering kernels are of size 3×3. The Max-pooling is performed over 2×2 pixel windows, with stride 2. The activation function for all hidden layers is the rectifier linear unit (ReLU). At the top of the VGG-16 architecture there are three fully connected layers. The size of the first two layers is 4096 and the size of the last is 1000 [153].

8.2 Experimental Design

In this research, we are using the VGG-16 pre-trained Convolutional Neural Network. The implementation of the algorithm was done in Matlab using the Neural Network Toolbox model and the VGG-16 Network support package in particular. For training and testing we used the DURHAM FACE database. The only one pre-processing step we did was the down-sampling the database images to a fixed size of 224×224, which is the standard input of VGG-16. Since the DURHAM FACE database consists of square images, no further cropping was required, and we did not process the images in any other way.

The transfer learning was conducted as follows; first, the raw images of the DURHAM FACE database were processed by the VGG-16 and the values of the second fully-connected layer (fc7) were obtained. They were used as a feature vector to train an SVM. In our implementations we used Matlab's *fitcecoc* function to train a binary SVM using the default parameters.

In some of our experiments, the imposter class was containing unprocessed im-

posters only, while in some other experiments it was containing sharpened imposters as well. In some pilot tests we trained a multi-class SVM, with the imposter class split into 5 different classes, depending on the amount of sharpening the imposters had received, but since the results were poor we dropped that idea.

We run both within-subject and cross-subject validation experiments. As DF database is a relatively small database containing face images from 21 individuals, we used images from all the 21 subjects in the within-subject training and test sets, while images from 15 individuals were used as the cross-subject training set and images from the remaining 6 individuals as the test set. In total we conducted four separate experiments using both within-subject and cross-subject protocols:

- Within-subject:

  We train with 788 client images and the same number of plain imposter images. We use 262 images from each of the 6 labels for testing.

  We train with 750 client images and 788 images from each of the 5 imposter labels. We test with 262 images from each of the 6 labels.

- Cross-subject:

  We train with 750 client images and the same number of plain imposter images. We use 300 images from each of the 6 labels for testing.

  We train with 750 client images and 750 images from each imposter label. We test with 300 images from each of the 6 labels.

8.3 Results

In Table 8.1, we report the accuracy rates for the various experiments evaluating the proposed algorithm based on the pre-trained VGG-16 as feature extractor and an SVM as classifier. We run four experiment, depending on whether the experimental design of training and testing is cross-subject or within-subject, and whether we used sharpened imposters for training. Notice that while the accuracy rates for the detection of an imposter are presented separately, in five different classes, it is still

a binary classification problem, that is, there are only two classes for training and testing, clients and imposters.

First, we notice that the proposed deep learning approach resulted in a considerably better performance than the previous approaches we presented in this thesis; the logistic regression and the shallow neural networks in particular. In the case of within subject validation in particular, the accuracy rates reached in most cases 100%. On the other hand, as expected, the cross subject setting poses a more challenging classification problem, in which case the use of a training set containing sharpened imposter images seems to be critical. Indeed, with cross-subject validation and training with imposter images only, the accuracy rates of the deep learning approach can be just slightly above 90% in some cases, and in the case of imposters sharpened by 8.0 the rate dips below 90%. In contrast, when we train with sharpened imposters too, the accuracy rates remains in most cases at 100%.

Prior to the main experiment, we conducted a pilot test in which we trained the SVM on six classes: client, sharp 0.0, 2.0, 4.0, 8.0 and 16.0. We noticed that training on these six classes resulted into a less robust liveness test. The accuracy rates for the cross-subject validation design were 93.33%, 76.67%, 36.67%, 60.00%, 83.33%, and 90.00% for client, sharp 0.0, 2.0, 4.0, 8.0, and 16.0, respectively, while in the within-subject validation design the accuracy rates from the same experiment were 95.24%, 95.24%, 90.48%, 100.00%, 100.00%, and 100.00%.

The proposed approach was implemented on a PC with an Intel® Core™ i3-3227U CPU running at 1.90 GHz and 4.00 GB RAM without parallel processing. The bulk of the detection time was consumed by the convolutional neural network as shown in Table 8.2.

8.4 Conclusions

In this chapter we evaluated the use of a relatively new machine learning technique, the transfer learning of a Convolutional Neural Network, which in many computer vision applications is providing the current state-of-the-art. We employed the widely used pre-trained VGG-16 deep network as a feature extractor and an SVM for

Table 8.1: Test results from four experimental designs: within-subject (top two) or cross-subject design (bottom two), and including or not sharpened imposters in the training set. In all cases, the test set contains both unprocessed and sharpened imposters.

Within-subject validation design with images from all 21 individuals in the training set. We train with clients and unprocessed imposters only.						
Labels	**client**	**sharp0.0**	**sharp2.0**	**sharp4.0**	**sharp8.0**	**sharp16.0**
Training set	788	788	-			
Test set	262 each					
Accuracy rate	100.00%	100.00%	100.00%	100.00%	98.09%	98.48%
Within-subject validation design with images from all 21 individuals in the training set. We train with clients, unprocessed imposters and sharpened imposters.						
Labels	**client**	**sharp0.0**	**sharp2.0**	**sharp4.0**	**sharp8.0**	**sharp16.0**
Training set	750	4725 images (avg 788 each)				
Test set	262 each					
Accuracy rate	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
Cross-subject validation design with images from 15 and 6 individuals for training and test, respectively. We train with clients and unprocessed imposters only.						
Labels	**client**	**sharp0.0**	**sharp2.0**	**sharp4.0**	**sharp8.0**	**sharp16.0**
Training set	750	750	-			
Test set	300 each					
Accuracy rate	90.91%	96.67%	93.00%	98.00%	86.33%	90.67%
Cross-subject validation design with images from 15 and 6 individuals for training and test, respectively. We train with clients, unprocessed imposters and sharpened imposters.						
Labels	**client**	**sharp0.0**	**sharp2.0**	**sharp4.0**	**sharp8.0**	**sharp16.0**
Training set	750	3750 images (750 each)				
Test set	300 each					
Accuracy rate	96.30%	100.00%	100.00%	100.00%	100.00%	100.00%

Table 8.2: Timings.

Exp1: Within-subject validation: 1,576 images (2 labels). Test set: 1,572		
VGG on training set	**VGG on test set**	**SVM training and testing**
∼15 hrs	∼15 hrs	32 secs
Exp2: Within-subject validation: 5,475 images (3 labels). Test set: 1,572		
VGG on training set	**VGG on test set**	**SVM training and testing**
∼54 hrs	∼15 hrs	52 secs
Exp3: Cross-subject validation: 1,500 images (2 labels). Test set: 1,800		
VGG on training set	**VGG on test set**	**SVM training and testing**
∼14.2 hrs	∼17.2 hrs	31 secs
Exp4: Cross-subject validation: 4,500 images (3 labels). Test set: 1,800		
VGG on training set	**VGG on test set**	**SVM training and testing**
∼43.6 hrs	∼17.2 hrs	47 secs

the classification, aiming at detecting malicious spoofing attacks using processed imposter images. The initial validation of the proposed approach shows that it can be an extremely robust liveness detection technique, even in a cross-subject validation setting. But in that case, training with sharpened imposters is required in order to achieve the high accuracy rates that approach 100%.

In the future, we would like to extend the DURHAM FACE database with imposter images processed in various other ways beyond plain sharpening and use it to fine tune a pre-trained Convolutional Neural Network. Further work on this direction will involve the augmentation of other publicly available databases with images from processed imposter attacks.

In another direction for future work, we will utilize other CNN architectures for transfer learning or for training classifiers from scratch. Examples of such architectures include AlexNet [99], GoogleNet [175] and ResNet [70]. The ResNet in particular, consisting of 152 layer of conv-relu-conv series, is an example of the current state-of-the-art very deep neural networks and we will consider adapting its architecture for our purpose of detecting processed image imposter attacks.

Chapter 9

Conclusions and Future Work

9.1 Introduction

Biometric security is the authentication of people's identity by an instant and automatic verification mechanism using user's physical characteristics to provide them with an access to the desired system. Nowadays, biometric security has become one of the most, if not the most, robust verification method. The various biometric traits can be either physiological or behavioural. Iris, fingerprint, face, and vein are considered as the most usable physiological biometrics while behavioural biometrics include activities such as gesturing, gait, and speech. Face recognition is a very popular choice in security systems, and recently is considered the most common identification system in commercial use.

Nevertheless, biometrics have been found to be vulnerable to malicious attacks. These spoofing attempts can either be *direct* or *indirect*. Direct attacks occur at the sensor level of the system, such attacks do not require any prior knowledge of the structure of the system. On the other hand, indirect attacks require some knowledge of the algorithmic methods employed the system.

Previous researches have shown that direct attacks can be deployed by using printed photos, or by displaying videos or images of the biometric modality in-front of the system's sensor. Our research focus on direct attacks to systems, and the main contribution is that we studied such direct attacks on face recognition systems assuming the use of processed imposter images.

9.2 Research Contributions

The contributions of our research documented in this thesis are summarized as follows:

1. **Evaluating the Resilience of Commercial Face Recognition Systems against Malicious Attacks.**

   In Chapter 4, we study the resilience of four widely used commercial face recognition systems against various attacks using imposter images. Two sessions were conducted for this experiment; the first session is studying whether the images used in the attack can be user's ID photos, photos found on-line, or any instant images taken on a mobile device. All of these four systems were found vulnerable to our suggested very crude attacks. In the second session, we studied the ability to gain access into the system by using photos of users taken from different distances, aiming at finding out whether longer distances between the face image and the sensor can result in less effective attacks to the systems.

   The second part of this contribution is to identify the image compression rate threshold, which still allows successful attacks. In a final experiment, zooming into the face when longer distance photos are used and then cropping the image, was found partially successful as an attacking attempt.

   The results of this contribution have been published in a paper entitled "Evaluating the Resilience of Face Recognition Systems Against Malicious Attacks" [130]. This work was been of importance to other researchers to support the motivation for their work by citing the fact that current commercial face recognition systems have been found vulnerable to direct attacks, which can be as basic as using highly compressed images, or images downloaded from on-line sources or social media.

2. **Resilience of Luminance based Liveness Tests under Attacks with Processed Imposter Images**

   The main contribution of Chapter 5 is to evaluate the sensitivity of the liveness detection test presented by Tan et al. [177] against processed imposter

image attacks. In a real life scenario, it is expected that the attackers would consider processing the images to be used for spoofing attempts if they know that would increase the effectiveness of the attacks. Thus, in this chapter, we demonstrated that the accuracy rate of the liveness detection algorithm proposed in [177] is affected by the use of processed imposter images. An existed well-known face spoofing database, the NUAA which has client and plain imposter datasets, was used to test the validity of our claim.

In our experiment, three main processing techniques were applied on a set of images; the Gaussian blur, the addition of salt and pepper noise, and finally the use of a sharpening filter. We showed that there is a considerable decrease in the performance of the Tan et al. liveness detection algorithm when sharpening the imposter images by various amounts. In particular, we showed that higher amounts of sharpening lead to worse performance of the anti-spoofing technique. In contrast, other image processing techniques such as blurring, adding noise, or even blurring a sharpened image resulted in an increase of the performance of the evaluated anti-spoofing method. The results of this contribution have been published in a paper entitled "Resilience of luminance based liveness tests under attacks with processed imposter images" [132].

3. **Designing a Facial Spoofing Database for Processed Image Attacks**
   After a thorough research, we found that there is no face spoofing database with processed imposter images. Hence, we decided to design a facial spoofing database for processed imposter image attacks. This database has been designed to include real live, recaptured photo images, and processed recaptured photo images.

   Designing such a database is a challenging task as there is a multitude of parameters to consider. These parameters can include the type of the camera to be used for capturing images, the illumination conditions during the photo-shooting sessions, the camera focus mechanism, or the printer to be used for producing imposters. In Chapter 6, we made an effort to understand these challenging issues by conducted brief pilot studies, and produced a competitive facial spoofing database. We tested this database on the Tan et al. liveness

test and found that, as expected, the test was performing worse with processed imposter images. The results of this contribution have been published in a paper entitled "Designing a facial spoofing database for processed image attacks" [131].

4. **Signal-Noise Analysis of Face Anti-spoofing Algorithms**
   After conducting research on the robustness of our chosen liveness detection algorithm, and found that the use of processed imposter images decreases its performance, in Chapter 7 we go further into the detail modelling of how the amount of sharpening affects performance. We modelled the outcomes of the experiment using beta distributions for the signal and the noise, overcoming certain limitations of the ROC curves.

5. **Transfer Learning for Face Liveness Detection**
   Finally, in Chapter 8 we run initial anti-spoofing experiments using transfer learning and Convolutional Neural Networks. Our first results verify that processed image attacks are more difficult to detect even when using high-end expensive machine learning techniques.

9.3 Limitations

1. **Variation of the data**
   As noted in Chapter 6, creating a new comprehensive face database is not a straightforward procedure, and a well designed plan is needed before starting collecting data. This planning includes considering certain experimental conditions and defining the parameters of the dataset. Due to the lack of experience, time, and human resources, we could not consider all such factors in the creation of our database. As a result, the value of some parameters could not be easily measured and qualified while others could not considered at all.

2. **Size of the facial spoofing database**
   Deep learning has become the best performing machine learning techniques in many applications. One limitation that prevents us from using deep neural

networks in liveness tests is the lack of a very large face spoofing database. This is a main challenge hindering further rapid progress in the face liveness detection area.

9.4 Future Work

In the future, our plans are to consider the followings:

1. **Extensions of the DURHAM FACE database**

   In the future, we aim at expanding the DURHAM FACE database to include more types of processed imposter images, and if possible reverse engineer the process of creating imposter images to make them look as close to the clients as possible.

   As we understand there is variety of face poses and expressions on social media, and other publicly available images and videos along with many different parameters from resolution to depth of focus and we intend to use them to inform the future of our database.

2. **Evaluation methods**

   There are hundreds of anti-spoofing algorithms, especially regarding face recognition. As it is impossible to consider all of these available methods, we aim at considering the most well-known and most robust of them. At the machine learning level, we aim at considering various feature extraction techniques and several classifiers.

3. **Liveness detection techniques**

   Our research highlighted the fact that even robust anti-spoofing algorithms are still vulnerable to relatively simple attacks such as using processed imposter images. In the future, we plan to work on developing stronger liveness tests that are robust to any amended imposters. Our aim is that such liveness tests will still not require any non-standard hardware such as the illuminated IR sensor required by Windows Hello.

Appendix A

Evaluating the Resilience of Commercial Face Recognition Systems

This appendix contains the consent form and the questionnaire, which were filled in by the participants for the experiment conducted to evaluate the resilience of commercial face recognition systems against malicious attacks in Chapter 4.

Participant No:……

Dear Sir, Madam,

Thank you for being a volunteering participant in my research study "**Evaluating the resilience Face Recognition systems against malicious attacks**". This experiment aims to test the security of different face recognition systems for different machines. You will be part of this experiment by participating in verifying your face to different systems, then you'll be taken an instant photo for your face through a mobile device, also you'll be asked to offer any ID photo for yourself and finally to provide a link to some of published on internet/social media (Google, Facebook, Twitter, Instagram, WhatsApp, etc.) photos for yourself. Those photos then will be used to gain access to our tested systems, then your photos will be resized several times to be used in the same experiment. While or after the experiment, kindly fill the attached form.

There are no direct risks to you by participating in the study, if you find any inconvenience during the test you may withdraw at any time. If you have any questions, or would like a copy of the project outline, please contact Luma Omar (Luma.omar@durham.ac.uk).

Best Regards,

Luma Omar

I have read the above document and I agree to participate in the study.

_____	_____	_____
Participant's Printed Name	Participant's Signature	Date (dd/mm/yyyy)

Participant No:……

Evaluating the resilience Face Recognition systems against malicious attacks

Gender: Male [ ] Female [ ] Age: _____

Your face has been verified to the systems Yes [ ] No [ ]

Instant Photos by the mobile from different distances:-

I. Instant Photos by the mobile from a close distance (approx 50 cm):

- Your instant image was successful to gain access to at least one system: Yes [ ] No [ ]
  - If yes, then what system(s):
    - ✓ KeyLemon Yes [ ] No [ ]
    - ✓ Android Face Unlock Yes [ ] No [ ]
    - ✓ Windows 10 Yes [ ] No [ ]

II. Instant Photos by the mobile from a medium distance (nearly 100 cm):

- Your instant image was successful to gain access to at least one system Yes [ ] No [ ]
  - If yes, then what system(s):-
    - ✓ KeyLemon Yes [ ] No [ ]
    - ✓ Android Face Unlock Yes [ ] No [ ]
    - ✓ Windows 10 Yes [ ] No [ ]

III. Instant Photos by the mobile from a medium distance (nearly 150 cm):

- Your instant image was successful to gain access to at least one system Yes [ ] No [ ]
  - If yes, then what system(s):-
    - ✓ KeyLemon Yes [ ] No [ ]
    - ✓ Android Face Unlock Yes [ ] No [ ]
    - ✓ Windows 10 Yes [ ] No [ ]

ID Photos:

- Your ID photo was successful to gain access to at least one system — Yes [ ] No [ ]
  - If yes, then what system(s):-
    - ✓ KeyLemon — Yes [ ] No [ ]
    - ✓ Android Face Unlock — Yes [ ] No [ ]
    - ✓ Windows 10 — Yes [ ] No [ ]

Photos on Internet/Social Media:

- At least one of your **Internet/Social Media** ant image was successful to gain access to at least one system — Yes [ ] No [ ]
  - If yes, then mention number of photos for each system:-
    - ✓ KeyLemon _____ out of _____
    - ✓ Android Face Unlock _____ out of _____
    - ✓ Windows 10 _____ out of _____

I thank you for your cooperation

Luma Omar

Appendix B

Durham Face Database

This appendix contains the summary and the consent form which have been given to the participants for the creation of the Durham Face database in Chapter 6.

B.1 DF Database Creation Project Summary

Creating a face image database for liveness tests

Lama Omar

The aim is to create a digital database consisting of digital images of genuine faces and 'imposter digital images', that is, digital images of genuine faces images printed on paper or displayed on a smartphone or tablet screen. In the Figure below you can see a typical example of a genuine face image and an imposter face image from the publicly available NUAA face image database for liveness tests.

Real **Imposter** **Real** **Imposter**

Figure: A typical example of a genuine face image (digital photo) and an imposter image (digital photo of a printout of the genuine face image) from the publicly available NUAA database for liveness tests.

This database is expected to be used by me and other researchers on evaluating and further developing liveness tests, that is, algorithms aiming at distinguishing between genuine face images and imposter face images.

B.2 DF Database Creation Consent Form

Participant No: ……

Dear participant,

Thank you for being a participant in my research project "**Creating a face image database for liveness tests**".

In this photo-shooting session, which will last for a maximum of 15 minutes, several photos of your face will be taken. These photos, the corresponding imposter images, and processed versions of them will be part of the database. Please read the attached summary of the project for more information on how the database will be created.

The face database will be publicly available for unrestricted access over the internet, but no information about your person (e.g. name, age, gender) will be attached to the images, or will become part of the database in any other way.

There are no direct risks to you by participating in the study, if you find any inconvenience during the test you may withdraw at any time. If you have any questions, or would like a copy of the project outline, please contact Luma Omar (luma.omar@durham.ac.uk).

At the end of the photo-shooting session, you will be given £5.00 as a thank you for participation.

Thank you for your participation,

Luma Omar

I have read the documents and I agree to participate in the research.

_____	_____	_____
Participant's Printed Name	Participant's Signature	Date (dd/mm/yyyy)

Bibliography

[1] About keylemon. http://www.keylemon.com, last accessed: July, 2015.

[2] Android FaceUnlock. http://www.androidcentral.com, last accessed: July, 2015.

[3] FaceLock for Apps. http://www.facelock.mobi/, last accessed: July, 2015.

[4] Luxand FaceSDK. http://www.luxand.com, last accessed: July, 2015.

[5] Abhyankar, A., and Schuckers, S. Integrating a wavelet based perspiration liveness check with fingerprint recognition. In *Pattern Recognition* (2009), vol. 42, Elsevier, pp. 452–464.

[6] Adler, A. Sample images can be independently restored from face recognition templates. In *Canadian Conference on Electrical and Computer Engineering* (2003), vol. 2, IEEE, pp. 1163–1166.

[7] Agarwal, A., Singh, R., and Vatsa, M. Face anti-spoofing using haralick features. In *IEEE 8th International Conference on Biometrics Theory, Applications and Systems* (2016), IEEE, pp. 1–6.

[8] Aggarwal, G., Chowdhury, A. R., and Chellappa, R. A system identification approach for video-based face recognition. In *Proceedings of the 17th International Conference on Pattern Recognition* (2004), vol. 4, IEEE, pp. 175–178.

[9] Aggarwal, G., Ratha, N. K., Jea, T.-Y., and Bolle, R. M. Gradient based textural characterization of fingerprints. In *2nd IEEE International*

Conference on Biometrics: Theory, Applications and Systems (2008), IEEE, pp. 1–5.

[10] Akhtar, Z., Micheloni, C., and Foresti, G. L. Biometric liveness detection: Challenges and research opportunities. In *IEEE Security & Privacy* (2015), vol. 13, IEEE, pp. 63–72.

[11] Akhtar, Z., Micheloni, C., Piciarelli, C., and Foresti, G. L. Mobio_livdet: Mobile biometric liveness detection. In *Advanced Video and Signal Based Surveillance (AVSS), 2014 11th IEEE International Conference on* (2014), IEEE, pp. 187–192.

[12] Alotaibi, A., and Mahmood, A. Enhancing computer vision to detect face spoofing attack utilizing a single frame from a replay video attack using deep learning. In *ICOIP* (2016), pp. 1–5.

[13] Anjos, A., Chakka, M., and Marcel, S. Motion-based counter-measures to photo attacks in face recognition. In *IET Biometrics* (2014), vol. 3, pp. 147–158.

[14] Anjos, A., and Marcel, S. Counter-measures to photo attacks in face recognition: A public database and a baseline. In *IJCB* (2011), pp. 1–7.

[15] Baccouche, M., Mamalet, F., Wolf, C., Garcia, C., and Baskurt, A. Sequential deep learning for human action recognition. In *International Workshop on Human Behavior Understanding* (2011), Springer, pp. 29–39.

[16] Ballard, L., Lopresti, D., and Monrose, F. Forgery quality and its implications for behavioral biometric security. In *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* (2007), vol. 37, IEEE, pp. 1107–1118.

[17] Bao, W., Li, H., Li, N., and Jiang, W. A liveness detection method for face recognition based on optical flow field. In *IEEE IASP* (2009), pp. 233–236.

[18] Bartlett, M., Movellan, J. R., and Sejnowski, T. Face recognition by independent component analysis. In *IEEE Transactions on Neural Networks* (2002), vol. 13, pp. 1450–1464.

[19] Bebis, G., Gyaourova, A., Singh, S., and Pavlidis, I. Face recognition by fusing thermal infrared and visible imagery. In *Image and Vision Computing* (2006), vol. 24, Elsevier, pp. 727–742.

[20] Belfiore, J. Making Windows 10 more personal and more secure with Windows Hello. http://blogs.windows.com/bloggingwindows/2015/03/17/making-windows-10-more-personal-and-more-secure-with-windows-hello, last accessed: July, 2015.

[21] Belhumeur, P. N., Hespanha, J. P., and Kriegman, D. J. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. In *IEEE Transactions on pattern analysis and machine intelligence* (1997), vol. 19, IEEE, pp. 711–720.

[22] Besbes, F., Trichili, H., and Solaiman, B. Multimodal biometric system based on fingerprint identification and iris recognition. In *3rd International Conference on Information and Communication Technologies: From Theory to Applications* (2008), IEEE, pp. 1–5.

[23] Beveridge, J. R., Zhang, H., Draper, B. A., Flynn, P. J., Feng, Z., Huber, P., Kittler, J., Huang, Z., Li, S., Li, Y., et al. Report on the fg 2015 video person recognition evaluation. In *11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition* (2015), vol. 1, IEEE, pp. 1–8.

[24] Bolle, R., and Pankanti, S. *Biometrics, Personal Identification in Networked Society: Personal Identification in Networked Society.* Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[25] Bowyer, K. W., Hollingsworth, K., and Flynn, P. J. Image understanding for iris biometrics: A survey. In *Computer vision and image understanding* (2008), vol. 110, Elsevier, pp. 281–307.

[26] Brzezinski, J. R., and Knafl, G. J. Logistic regression modeling for context-based classification. In *Tenth International Workshop on Database and Expert Systems Applications, 1999. Proceedings* (1999), IEEE, pp. 755–759.

[27] Bu, W., Wu, X., and Gao, E. Hand vein recognition based on orientation of lbp. In *SPIE Defense, Security, and Sensing* (2012), International Society for Optics and Photonics, pp. 83711Y–83711Y.

[28] Chakraborty, S., and Das, D. An overview of face liveness detection. In *arXiv preprint arXiv:1405.2227* (2014).

[29] Chan, P. P., Liu, W., Chen, D., Yeung, D. S., Zhang, F., Wang, X., and Hsu, C.-C. Face liveness detection using a flash against 2d spoofing attack. In *IEEE Transactions on Information Forensics and Security* (2018), vol. 13, IEEE, pp. 521–534.

[30] Chang, M., Yih, W., and Meek, C. A logistic regression model for detecting prominences. In *ACM SIGKDD International conference on Knowledge Discovery and Data mining* (1996), pp. 2443–2445.

[31] Chetty, G., and Wagner, M. Multi-level liveness verification for face-voice biometric authentication. In *Biometric Consortium Conference, 2006 Biometrics Symposium: Special Session on Research at the* (2006), IEEE, pp. 1–6.

[32] Chikkerur, S., Pankanti, S., Jea, A., Ratha, N., and Bolle, R. Fingerprint representation using localized texture features. In *18th International Conference on Pattern Recognition* (2006), vol. 4, IEEE, pp. 521–524.

[33] Chingovska, I., Anjos, A., and Marcel, S. On the effectiveness of local binary patterns in face anti-spoofing. In *BIOSIG* (2012), IEEE, pp. 1–7.

[34] Chopra, S., Hadsell, R., and LeCun, Y. Learning a similarity metric discriminatively, with application to face verification. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2005), vol. 1, IEEE, pp. 539–546.

[35] Choudhury, T., Clarkson, B., Jebara, T., and Pentland, A. Multimodal person recognition using unconstrained audio and video. In *Proceedings, International Conference on Audio-and Video-Based Person Authentication* (1999), pp. 176–181.

[36] Cinbis, R. G., Verbeek, J., and Schmid, C. Unsupervised metric learning for face identification in tv video. In *IEEE International Conference on Computer Vision* (2011), IEEE, pp. 1559–1566.

[37] Coetzee, L., and Botha, E. C. Fingerprint recognition in low quality images. In *Pattern recognition* (1993), vol. 26, Elsevier, pp. 1441–1460.

[38] Dahiya, N., and Kant, C. Biometrics security concerns. In *Second International Conference on Advanced Computing & Communication Technologies* (2012), IEEE, pp. 297–302.

[39] Daugman, J. Recognizing people by their iris patterns. In *Information Security Technical Report* (1998), vol. 3, Elsevier, pp. 33–39.

[40] Daugman, J. Demodulation by complex-valued wavelets for stochastic pattern recognition. In *International Journal of Wavelets, Multiresolution and Information Processing* (2003), vol. 1, World Scientific, pp. 1–17.

[41] Daugman, J. How iris recognition works. In *IEEE Transactions on circuits and systems for video technology* (2004), vol. 14, IEEE, pp. 21–30.

[42] Daugman, J. Iris recognition and anti-spoofing countermeasures. In *7-th International Biometrics conference* (2004).

[43] Daugman, J. New methods in iris recognition. In *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* (2007), vol. 37, IEEE, pp. 1167–1175.

[44] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition* (2009), IEEE, pp. 248–255.

[45] Derakhshani, R., Schuckers, S. A., Hornak, L. A., and O'Gorman, L. Determination of vitality from a non-invasive biomedical measurement for use in fingerprint scanners. In *Pattern recognition* (2003), vol. 36, Elsevier, pp. 383–396.

[46] Di Martino, J. M., Qiu, Q., Nagenalli, T., and Sapiro, G. Liveness detection using implicit 3d features. In *arXiv preprint arXiv:1804.06702* (2018).

[47] Ding, C., and Tao, D. A comprehensive survey on pose-invariant face recognition. In *ACM Transactions on intelligent systems and technology* (2016), vol. 7, ACM, p. 37.

[48] Ding, Y., Zhuang, D., and Wang, K. A study of hand vein recognition method. In *IEEE International Conference Mechatronics and Automation* (2005), vol. 4, IEEE, pp. 2106–2110.

[49] Eum, S., Suhr, J. K., and Kim, J. Face recognizability evaluation for atm applications with exceptional occlusion handling. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops* (2011), IEEE, pp. 82–89.

[50] Faundez-Zanuy, M. Biometric security technology. In *IEEE Aerospace and Electronic Systems Magazine* (2006), vol. 21, IEEE, pp. 15–26.

[51] Feng, J., Jain, A. K., and Ross, A. Fingerprint alteration. In *submitted to IEEE TIFS* (2009).

[52] Flom, L., and Safir, A. Iris recognition system, 1987. US Patent 4,641,349.

[53] Foygel, R., and Mackey, L. Corrupted sensing: Novel guarantees for separating structured signals. In *IEEE Transactions on Information Theory* (2014), vol. 60, IEEE, pp. 1223–1247.

[54] Friedman, J., Hastie, T., Tibshirani, R., et al. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). In *The annals of statistics* (2000), vol. 28, Institute of Mathematical Statistics, pp. 337–407.

[55] Frischholz, R. W., and Dieckmann, U. Biold: a multimodal biometric identification system. In *Computer* (2000), vol. 33, IEEE, pp. 64–68.

[56] Galbally, J., Fierrez, J., and Ortega-Garcia, J. Bayesian hill-climbing attack and its application to signature verification. In *Advances in Biometrics* (2007), Springer, pp. 386–395.

[57] Galbally, J., Fierrez, J., Ortega-Garcia, J., McCool, C., and Marcel, S. Hill-climbing attack to an eigenface-based face verification system. In *BIdS* (2009), pp. 1–6.

[58] Galbally, J., Marcel, S., and Fierrez, J. Image quality assessment for fake biometric detection: Application to iris, fingerprint, and face recognition. In *IEEE Trans. on Image Processing* (2014), vol. 23, IEEE, pp. 710–724.

[59] Galbally, J., McCool, C., Fierrez, J., Marcel, S., and Ortega-Garcia, J. On the vulnerability of face verification systems to hill-climbing attacks. In *Pattern Recognition* (2010), vol. 43, Elsevier, pp. 1027–1038.

[60] Galbally, J., and Satta, R. Biometric sensor interoperability: A case study in 3d face recognition. In *ICPRAM* (2016), pp. 199–204.

[61] Girshick, R. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (2015), pp. 1440–1448.

[62] Girshick, R., Donahue, J., Darrell, T., and Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2014), pp. 580–587.

[63] Gomez-Barrero, M., Galbally, J., and Fierrez, J. Efficient software attack to multimodal biometric systems and its application to face and iris fusion. In *Pattern Recognition Letters* (2014), vol. 36, Elsevier, pp. 243–253.

[64] Gross, R., and Shi, J. The cmu motion of body (mobo) database. Tech. rep., 2001.

[65] Han, W.-Y., and Lee, J.-C. Palm vein recognition using adaptive gabor filter. In *Expert Systems with Applications* (2012), vol. 39, Elsevier, pp. 13225–13234.

[66] Hand, D. J. Measuring classifier performance: a coherent alternative to the area under the roc curve. In *Machine learning* (2009), vol. 77, Springer, pp. 103–123.

[67] Hand, D. J., and Anagnostopoulos, C. A better beta for the h measure of classification performance. In *Pattern Recognition Letters* (2014), vol. 40, Elsevier, pp. 41–46.

[68] Hawthorne, M. R. *Fingerprints: analysis and understanding*. CRC Press, 2008.

[69] He, K., Zhang, X., Ren, S., and Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision* (2015), pp. 1026–1034.

[70] He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 770–778.

[71] He, M., Horng, S.-J., Fan, P., Run, R.-S., Chen, R.-J., Lai, J.-L., Khan, M. K., and Sentosa, K. O. Performance evaluation of score level fusion in multimodal biometric systems. In *Pattern Recognition* (2010), vol. 43, Elsevier, pp. 1789–1800.

[72] He, X., Yan, S., Hu, Y., Niyogi, P., and Zhang, H.-J. Face recognition using laplacianfaces. In *IEEE transactions on pattern analysis and machine intelligence* (2005), vol. 27, IEEE, pp. 328–340.

[73] Hong, L., and Jain, A. Integrating faces and fingerprints for personal identification. In *IEEE transactions on pattern analysis and machine intelligence* (1998), vol. 20, IEEE, pp. 1295–1307.

[74] Hu, J., Lu, J., and Tan, Y.-P. Discriminative deep metric learning for face verification in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2014), pp. 1875–1882.

[75] Huang, B., Dai, g., Li, R., Tang, D., and Li, W. Finger-vein authentication based on wide line detector and pattern normalization. In *20th International Conference on Pattern Recognition* (2010), IEEE, pp. 1269–1272.

[76] Huang, G. B., Lee, H., and Learned-Miller, E. Learning hierarchical representations for face verification with convolutional deep belief networks. In *IEEE Conference on Computer Vision and Pattern Recognition* (2012), IEEE, pp. 2518–2525.

[77] Huang, G. B., Ramesh, M., Berg, T., and Learned-Miller, E. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Tech. rep., Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[78] Huang, P. S., Harris, C. J., and Nixon, M. S. Recognising humans by gait via parametric canonical space. In *Artificial Intelligence in Engineering* (1999), vol. 13, Elsevier, pp. 359–366.

[79] Huang, X., Ti, C., Hou, Q.-z., Tokuta, A., and Yang, R. An experimental study of pupil constriction for liveness detection. In *IEEE Workshop on Applications of Computer Vision* (2013), IEEE, pp. 252–258.

[80] Jain, A., Ross, A., and Prabhakar, S. Fingerprint matching using minutiae and texture features. In *International Conference on Image Processing, 2001. Proceedings* (2001), vol. 3, IEEE, pp. 282–285.

[81] Jain, A. K., Ross, A., and Pankanti, S. Biometrics: a tool for information security. In *IEEE transactions on information forensics and security* (2006), vol. 1, IEEE, pp. 125–143.

[82] Jain, A. K., Ross, A., and Prabhakar, S. An introduction to biometric recognition. In *IEEE Transactions on circuits and systems for video technology* (2004), vol. 14, IEEE, pp. 4–20.

[83] Jani, R., and Agrawal, N. A proposed framework for enhancing security in fingerprint and finger-vein multimodal biometric recognition. In *International Conference on Machine Intelligence and Research Advancement* (2013), IEEE, pp. 440–444.

[84] Jee, H.-K., Jung, S.-U., and Yoo, J.-H. Liveness detection for embedded face recognition system. In *International Journal of Biological and Medical Sciences* (2006), vol. 1, pp. 235–238.

[85] Ji, S., Xu, W., Yang, M., and Yu, K. 3d convolutional neural networks for human action recognition. In *IEEE transactions on pattern analysis and machine intelligence* (2013), vol. 35, IEEE, pp. 221–231.

[86] Jurafsky, D. Speech and language processing: An introduction to natural language processing. In *Computational linguistics, and speech recognition* (2000), Prentice Hall.

[87] Kanematsu, M., Takano, H., and Nakamura, K. Highly reliable liveness detection method for iris recognition. In *SICE, 2007 Annual Conference* (2007), IEEE, pp. 361–364.

[88] Karson, C. N. Spontaneous eye-blink rates and dopaminergic systems. In *Brain* (1983), vol. 106, Oxford University Press, pp. 643–653.

[89] Kim, G., Eum, S., Kyu Suhr, J., Ik Kim, D., Ryoung Park, K., and Kim, J. Face liveness detection based on texture and frequency analyses. In *5th IAPR International Conference on Biometrics* (2012), pp. 67–72.

[90] Kim, S., Ban, Y., and Lee, S. Face liveness detection using a light field camera. In *Sensors* (2014), vol. 14, Multidisciplinary Digital Publishing Institute, pp. 22471–22499.

[91] Kim, S., Yu, S., Kim, K., Ban, Y., and Lee, S. Face liveness detection using variable focusing. In *International Conference on Biometrics* (2013), IEEE, pp. 1–6.

[92] Kim, W., Suh, S., and Han, J.-J. Face liveness detection from a single image via diffusion speed model. In *IEEE transactions on Image processing* (2015), vol. 24, IEEE, pp. 2456–2465.

[93] Koh, K., Kim, S.-J., and Boyd, S. An interior-point method for large-scale l1-regularized logistic regression. In *Journal of Machine learning research* (2007), vol. 8, pp. 1519–1555.

[94] Kohli, N., Yadav, D., Vatsa, M., and Singh, R. Revisiting iris recognition with color cosmetic contact lenses. In *International Conference on Biometrics* (2013), IEEE, pp. 1–7.

[95] Kollreider, K., Fronthaler, H., and Bigun, J. Evaluating liveness by face images and the structure tensor. In *Fourth IEEE Workshop on Automatic Identification Advanced Technologies* (2005), IEEE, pp. 75–80.

[96] Kollreider, K., Fronthaler, H., and Bigun, J. Non-intrusive liveness detection by face images. In *Image and Vision Computing* (2009), vol. 27, Elsevier, pp. 233–244.

[97] Kollreider, K., Fronthaler, H., Faraj, M. I., and Bigun, J. Real-time face detection and motion analysis with application in liveness assessment. In *IEEE Transactions on Information Forensics and Security* (2007), vol. 2, IEEE, pp. 548–558.

[98] Komulainen, J., Hadid, A., and Pietikainen, M. Context based face anti-spoofing. In *Sixth International Conference on Biometrics: Theory, Applications and Systems* (2013), IEEE, pp. 1–8.

[99] Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (2012), pp. 1097–1105.

[100] Lagorio, A., Tistarelli, M., Cadoni, M., Fookes, C., and Sridharan, S. Liveness detection based on 3d face shape analysis. In *International Workshop on Biometrics and Forensics* (2013), IEEE, pp. 1–4.

[101] Lai, C., and Tai, C. A smart spoofing face detector by display features analysis. In *Sensors* (2016), vol. 16, Multidisciplinary Digital Publishing Institute, p. 1136.

[102] Lee, E. C., Park, K. R., and Kim, J. Fake iris detection by using purkinje image. In *International Conference on Biometrics* (2006), Springer, pp. 397–403.

[103] Lee, J.-C. A novel biometric system based on palm vein image. In *Pattern Recognition Letters* (2012), vol. 33, Elsevier, pp. 1520–1528.

[104] Lee, K.-C., Ho, J., Yang, M.-H., and Kriegman, D. Video-based face recognition using probabilistic appearance manifolds. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2003), vol. 1, pp. I–313–I–320 vol.1.

[105] Lee, L., and Grimson, W. E. L. Gait analysis for recognition and classification. In *Fifth IEEE International Conference on Automatic Face and Gesture Recognition* (2002), IEEE, pp. 155–162.

[106] Li, J., Wang, Y., Tan, T., and Jain, A. K. Live face detection based on the analysis of fourier spectra. In *Defense and Security* (2004), pp. 296–303.

[107] Li, L., Feng, X., Boulkenafet, Z., Xia, Z., Li, M., and Hadid, A. An original face anti-spoofing approach using partial convolutional neural network. In *IPTA* (2016), IEEE, pp. 1–6.

[108] Li, Y., Xu, K., Yan, Q., Li, Y., and Deng, R. H. Understanding osn-based facial disclosure against face authentication systems. In *Proceedings of the 9th ACM symposium on Information, computer and communications security* (2014), ACM, pp. 413–424.

[109] Liao, J., and Chin, K.-V. Logistic regression for disease classification using microarray data: model selection in a large p and small n case. In *Bioinformatics* (2007), vol. 23, Oxford University Press, pp. 1945–1951.

[110] Little, J., and Boyd, J. Recognizing people by their gait: the shape of motion. In *Videre: Journal of computer vision research* (1998), vol. 1, pp. 1–32.

[111] Liu, J., Chen, J., and Ye, J. Large-scale sparse logistic regression. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (2009), ACM, pp. 547–556.

[112] Liu, X., and Chen, T. Video-based face recognition using adaptive hidden markov models. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2003), vol. 1, pp. I–340–I–345.

[113] Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 3730–3738.

[114] Lu, C., and Tang, X. Surpassing human-level face verification performance on lfw with gaussianface. In *AAAI* (2015), pp. 3811–3819.

[115] Lucena, O., Junior, A., Moia, V., Souza, R., Valle, E., and Lotufo, R. Transfer learning using convolutional neural networks for face anti-spoofing. In *International Conference Image Analysis and Recognition* (2017), Springer, pp. 27–34.

[116] Määttä, J., Hadid, A., and Pietikäinen, M. Face spoofing detection from single images using micro-texture analysis. In *IJCB* (2011), IEEE, pp. 1–7.

[117] Määttä, J., Hadid, A., and Pietikainen, M. Face spoofing detection from single images using texture and local shape analysis. In *IET Biometrics* (2012), vol. 1, pp. 3–10.

[118] Macmillan, N. A., and Douglas Creelman, C. *Detection Theory: A User's Guide*. Taylor & Francis, 2004.

[119] Maghbouleh, A. A logistic regression model for detecting prominences. In *Fourth International Conference on Spoken Language* (1996), vol. 4, IEEE, pp. 2443–2445.

[120] Maltoni, D., Maio, D., Jain, A., and Prabhakar, S. *Handbook of fingerprint recognition*. 2009.

[121] Man, J., and Bhanu, B. Individual recognition using gait energy image. In *IEEE transactions on pattern analysis and machine intelligence* (2006), vol. 28, IEEE, pp. 316–322.

[122] Marasco, E., and Ross, A. A survey on antispoofing schemes for fingerprint recognition systems. In *ACM Computing Surveys (CSUR)* (2015), vol. 47, ACM, p. 28.

[123] Martinez-Diaz, M., Fierrez-Aguilar, J., Alonso-Fernandez, F., Ortega-García, J., and Siguenza, J. Hill-climbing and brute-force attacks on biometric systems: A case study in match-on-card fingerprint verification. In *40th Annual IEEE International Carnahan Conferences Security Technology* (2006), IEEE, pp. 151–159.

[124] Matsumoto, T., Matsumoto, H., Yamada, K., and Hoshino, S. Impact of artificial gummy fingers on fingerprint systems. In *Proceedings of SPIE* (2002), vol. 4677, pp. 275–289.

[125] Menotti, D., Chiachia, G., Pinto, A., Schwartz, W. R., Pedrini, H., Falcão, A. X., and Rocha, A. Deep representations for iris, face, and fingerprint spoofing detection. In *IEEE Transactions on Information Forensics and Security* (2015), vol. 10, IEEE, pp. 864–879.

[126] Miura, N., Nagasaka, A., and Miyatake, T. Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. In *Machine Vision and Applications* (2004), vol. 15, Springer, pp. 194–203.

[127] Moayer, B., and Fu, K.-S. A tree system approach for fingerprint pattern recognition. In *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1986), no. 3, IEEE, pp. 376–387.

[128] Moriyama, T., Kanade, T., Cohn, J. F., Xiao, J., Ambadar, Z., Gao, J., and Imamura, H. Automatic recognition of eye blinking in spontaneously occurring behavior. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on* (2002), vol. 4, IEEE, pp. 78–81.

[129] Nogueira, R. F., de Alencar Lotufo, R., and Machado, R. C. Evaluating software-based fingerprint liveness detection using convolutional networks and local binary patterns. In *IEEE Workshop Proceedings on Biometric Measurements and Systems for Security and Medical Applications* , (2014), IEEE, pp. 22–29.

[130] Omar, L., and Ivrissimtzis, I. Evaluating the resilience of face recognition systems against malicious attacks. In *BMVW* (2015), pp. 5.1–5.9.

[131] Omar, L., and Ivrissimtzis, I. Designing a facial spoofing database for processed image attacks. In *ICDP* (2016), IET, pp. 5(6.) – 5(6.)(1).

[132] Omar, L., and Ivrissimtzis, I. Resilience of luminance based liveness tests under attacks with processed imposter images. In *WSCG* (2016), pp. 79–82.

[133] Oren, M., and Nayar, S. K. Generalization of the lambertian model and implications for machine vision. In *International Journal of Computer Vision* (1995), vol. 14, Springer, pp. 227–251.

[134] Ouyang, Z., Feng, J., Su, F., and Cai, A. Fingerprint matching with rotation-descriptor texture features. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on* (2006), vol. 4, IEEE, pp. 417–420.

[135] Owen, C. E. B. Parameter estimation for the beta distribution. Master's thesis, Brigham Young University, 2008.

[136] Pacut, A., and Czajka, A. Aliveness detection for iris biometrics. In *40th Annual IEEE International Carnahan Conferences Security Technology* (2006), IEEE, pp. 122–129.

[137] Pan, G., Sun, L., Wu, Z., and Lao, S. Eyeblink-based anti-spoofing in face recognition from a generic webcamera. In *ICCV* (2007), IEEE, pp. 1–8.

[138] Pan, G., Sun, L., Wu, Z., and Wang, Y. Monocular camera-based face liveness detection by combining eyeblink and scene context. In *Telecommunication Systems* (2011), vol. 47, Springer, pp. 215–225.

[139] Pan, G., Wu, Z., and Sun, L. Liveness detection for face recognition. In *Recent advances in face recognition*. InTech, 2008.

[140] Pan, S. J., and Yang, Q. A survey on transfer learning. In *IEEE Transactions on knowledge and data engineering* (2010), vol. 22, IEEE, pp. 1345–1359.

[141] Peixoto, B., Michelassi, C., and Rocha, A. Face liveness detection under bad illumination conditions. In *IEEE ICIP* (2011), pp. 3557–3560.

[142] Pentland, A., Moghaddam, B., and Starner, T. View-based and modular eigenspaces for face recognition. In *CVPR* (1994), pp. 84–91.

[143] Phillips, P. J., Flynn, P. J., Scruggs, T., Bowyer, K. W., Chang, J., Hoffman, K., Marques, J., Min, J., and Worek, W. Overview of the face recognition grand challenge. In *IEEE computer society conference*

on Computer vision and pattern recognition, 2005. CVPR 2005 (2005), vol. 1, IEEE, pp. 947–954.

[144] Poursheikhali Asgary, M., Jahandideh, S., Abdolmaleki, P., and Kazemnejad, A. Analysis and identification of β-turn types using multinomial logistic regression and artificial neural network. In *Bioinformatics* (2007), vol. 23, Oxford University Press, pp. 3125–3130.

[145] Raghavendra, R., and Busch, C. Robust scheme for iris presentation attack detection using multiscale binarized statistical image features. In *IEEE Transactions on Information Forensics and Security* (2015), vol. 10, IEEE, pp. 703–715.

[146] Rao, T. C. M. Feature extraction for fingerprint classification. In *Pattern Recognition* (1976), vol. 8, Elsevier, pp. 181–192.

[147] Ratha, N. K., Chen, S., and Jain, A. K. Adaptive flow orientation-based feature extraction in fingerprint images. In *Pattern Recognition* (1995), vol. 28, Elsevier, pp. 1657–1672.

[148] Ratha, N. K., Connell, J. H., and Bolle, R. M. An analysis of minutiae matching strength. In *International Conference on Audio-and Video-Based Biometric Person Authentication* (2001), Springer, pp. 223–228.

[149] Razavian, A. S., Azizpour, H., Sullivan, J., and Carlsson, S. Cnn features off-the-shelf: an astounding baseline for recognition. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops* (2014), IEEE, pp. 512–519.

[150] Ren, S., He, K., Girshick, R., and Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems* (2015), pp. 91–99.

[151] Ross, A., and Jain, A. Information fusion in biometrics. In *Pattern recognition letters* (2003), vol. 24, Elsevier, pp. 2115–2125.

[152] Ruiz-Albacete, V., Tome-Gonzalez, P., Alonso-Fernandez, F., Galbally, J., Fiérrez-Aguilar, J., and Ortega-Garcia, J. Direct attacks using fake images in iris verification. In *BIOID* (2008), Springer, pp. 181–190.

[153] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. In *International Journal of Computer Vision* (2015), vol. 115, Springer, pp. 211–252.

[154] Ryu, C., Kong, S. G., and Kim, H. Enhancement of feature extraction for low-quality fingerprint images using stochastic resonance. In *Pattern Recognition Letters* (2011), vol. 32, Elsevier, pp. 107–113.

[155] Sarkar, S., Phillips, P. J., Liu, Z., Vega, I. R., Grother, P., and Bowyer, K. W. The humanid gait challenge problem: Data sets, performance, and analysis. In *IEEE transactions on pattern analysis and machine intelligence* (2005), vol. 27, IEEE, pp. 162–177.

[156] Sartor, M. A., Leikauf, G. D., and Medvedovic, M. Lrpath: a logistic regression approach for identifying enriched biological groups in gene expression data. In *Bioinformatics* (2008), vol. 25, Oxford University Press, pp. 211–217.

[157] Schroff, F., Kalenichenko, D., and Philbin, J. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 815–823.

[158] Schuckers, M. E. Using the beta-binomial distribution to assess performance of a biometric identification device. In *International Journal of Image and Graphics* (2003), vol. 3, World Scientific, pp. 523–529.

[159] Schuckers, S., and Abhyankar, A. Detecting liveness in fingerprint scanners using wavelets: Results of the test dataset. In *Biometric Authentication* (2004), Springer, pp. 100–110.

[160] Schuckers, S., Hornak, L., Norman, T., Derakhshani, R., and Parthasaradhi, S. Issues for liveness detection in biometrics. In *Proceedings of Biometric Consortium Conference. IEEE, New York* (2002).

[161] Simonyan, K., and Zisserman, A. Very deep convolutional networks for large-scale image recognition. In *arXiv preprint arXiv:1409.1556* (2014).

[162] Singh, A. K., Joshi, P., and Nandi, G. C. Face recognition with liveness detection using eye and mouth movement. In *IEEE International Conference on Signal Propagation and Computer Technology* (2014), pp. 592–597.

[163] Sirovich, L., and Kirby, M. Low-dimensional procedure for the characterization of human faces. In *Journal of the Optical Society of America A* (1987), vol. 4, IEEE, pp. 519–524.

[164] Sivic, J., Everingham, M., and Zisserman, A. Person spotting: video shot retrieval for face sets. In *Image and Video Retrieval* (2005), Springer, pp. 592–592.

[165] Sivic, J., Everingham, M., and Zisserman, A. who are you?-learning person specific classifiers from video. In *IEEE Conference on Computer Vision and Pattern Recognition* (2009), IEEE, pp. 1145–1152.

[166] Socolinsky, D., Selinger, A., and Neuheisel, J. Face recognition with visible and thermal infrared imagery. In *Computer Vision and Image Understanding* (2003), vol. 91, Elsevier, pp. 72–114.

[167] Soviany, S., and Jurian, M. Multimodal biometric securing methods for informatic systems. In *34th International Spring Seminar on Electronics Technology* (2011), IEEE, pp. 447–450.

[168] Steiner, H., Sporrer, S., Kolb, A., and Jung, N. Design of an active multispectral swir camera system for skin detection and face verification. In *Journal of Sensors* (2015), vol. 2016, Hindawi Publishing Corporation.

[169] Stén, A., Kaseva, A., and Virtanen, T. Fooling fingerprint scanners-biometric vulnerabilities of the precise biometrics 100 sc scanner. In *Proceedings of 4th Australian Information Warfare and IT Security Conference* (2003), vol. 2003, pp. 333–340.

[170] Sun, L., Pan, G., Wu, Z., and Lao, S. Blinking-based live face detection using conditional random fields. In *Advances in Biometrics* (2007), Springer, pp. 252–260.

[171] Sun, Y., Chen, Y., Wang, X., and Tang, X. Deep learning face representation by joint identification-verification. In *Advances in neural information processing systems* (2014), pp. 1988–1996.

[172] Sun, Y., Wang, X., and Tang, X. Hybrid deep learning for face verification. In *Proceedings of the IEEE International Conference on Computer Vision* (2013), pp. 1489–1496.

[173] Sun, Y., Wang, X., and Tang, X. Deep learning face representation from predicting 10,000 classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2014), pp. 1891–1898.

[174] Sundaresan, A., RoyChowdhury, A., and Chellappa, R. A hidden markov model based framework for recognition of humans from gait sequences. In *International Conference on Image Processing* (2003), vol. 2, IEEE, pp. II–93.

[175] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 1–9.

[176] Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2014), pp. 1701–1708.

[177] Tan, X., Li, Y., Liu, J., and Jiang, L. Face liveness detection from a single image with sparse low rank bilinear discriminative model. In *ECCV* (2010), pp. 504–517.

[178] Tian, Y.-I., Kanade, T., and Cohn, J. F. Recognizing action units for facial expression analysis. In *IEEE Transactions on pattern analysis and machine intelligence* (2001), vol. 23, IEEE, pp. 97–115.

[179] Tsubota, K. Tear dynamics and dry eye. In *Progress in Retinal and Eye Research* (1998), pp. 565–596.

[180] Turk, M., and Pentland, A. Eigenfaces for recognition. In *3* (1991), MIT Press, pp. 71–86.

[181] Turk, M. A., and Pentland, A. P. Face recognition using eigenfaces. In *CVPR* (1991), IEEE, pp. 586–591.

[182] Uludag, U., and Jain, A. K. Attacks on biometric systems: a case study in fingerprints. In *Proceedings of SPIE* (2004), vol. 5306, pp. 622–633.

[183] Valencia, V., and Horn, C. Biometric liveness testing. In *Biometrics* (2003), New York: Osborne McGraw Hill, pp. 139–149.

[184] Van der Putte, T., and Keuning, J. Biometrical fingerprint recognition: dont get your fingers burned. In *Smart Card Research and Advanced Applications* (2000), Springer, pp. 289–303.

[185] Verma, M., Majumdar, A. K., and Chatterjee, B. Edge detection in fingerprints. In *Pattern Recognition* (1987), vol. 20, Elsevier, pp. 513–523.

[186] Wang, C., Li, K., Wu, Z., and Zhao, Q. A dcnn based fingerprint liveness detection algorithm with voting strategy. In *Chinese Conference on Biometric Recognition* (2015), Springer, pp. 241–249.

[187] Wang, L., Leedham, G., and Cho, D. S.-Y. Minutiae feature analysis for infrared hand vein pattern biometrics. In *Pattern recognition* (2008), vol. 41, Elsevier, pp. 920–929.

[188] Wang, L., Qiao, Y., and Tang, X. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 4305–4314.

[189] Wang, T., Yang, J., Lei, Z., Liao, S., and Li, S. Z. Face liveness detection using 3d structure recovered from a single camera. In *International Conference on Biometrics* (2013), IEEE, pp. 1–6.

[190] Wang, Y., Zhang, K., and Shark, L.-K. Personal identification based on multiple keypoint sets of dorsal hand vein images. In *IET Biometrics* (2014), vol. 3, IET, pp. 234–245.

[191] Wei, Z., Qiu, X., Sun, Z., and Tan, T. Counterfeit iris detection based on texture analysis. In *19th International Conference on Pattern Recognition* (2008), IEEE, pp. 1–4.

[192] Wen, D., Han, H., and Jain, A. K. Face spoof detection with image distortion analysis. In *IEEE Transactions on Information Forensics and Security* (2015), vol. 10, IEEE, pp. 746–761.

[193] Wildes, R. P. Iris recognition: an emerging biometric technology. In *Proceedings of the IEEE* (1997), vol. 85, IEEE, pp. 1348–1363.

[194] Wilson, C. L., Candela, G. T., and Watson, C. I. Neural network fingerprint classification. In *Journal of Artificial Neural Networks* (1994), vol. 1, pp. 203–228.

[195] Wolf, L., Hassner, T., and Maoz, I. Face recognition in unconstrained videos with matched background similarity. In *IEEE Conference on Computer Vision and Pattern Recognition* (2011), IEEE, pp. 529–534.

[196] Woodward, J. D. Biometrics: Privacy's foe or privacy's friend? In *Proceedings of the IEEE* (1997), vol. 85, IEEE, pp. 1480–1492.

[197] Wright, J., Yang, A. Y., Ganesh, A., Sastry, S. S., and Ma, Y. Robust face recognition via sparse representation. In *IEEE transactions on pattern analysis and machine intelligence* (2009), vol. 31, IEEE, pp. 210–227.

[198] Xu, Y., Price, T., Frahm, J.-M., and Monrose, F. Virtual u: Defeating face liveness detection by building virtual models from your public photos.

[199] Yadav, D., Kohli, N., Doyle, J. S., Singh, R., Vatsa, M., and Bowyer, K. W. Unraveling the effect of textured contact lenses on iris recognition. In *IEEE Transactions on Information Forensics and Security* (2014), vol. 9, IEEE, pp. 851–862.

[200] Yang, F., and Ma, B. A new mixed-mode biometrics information fusion based-on fingerprint, hand-geometry and palm-print. In *Fourth International Conference on Image and Graphics* (2007), IEEE, pp. 689–693.

[201] Yang, J., Lei, Z., and Li, S. Z. Learn convolutional neural network for face anti-spoofing. In *arXiv:1408.5601* (2014).

[202] Yang, J., Lei, Z., Liao, S., and Li, S. Z. Face liveness detection with component dependent descriptor. In *International Conference on Biometrics* (2013), pp. 1–6.

[203] Yang, J., and Li, X. Efficient finger vein localization and recognition. In *20th International Conference on Pattern Recognition* (2010), IEEE, pp. 1148–1151.

[204] Yao, Y., Marcialis, G. L., Pontil, M., Frasconi, P., and Roli, F. Combining flat and structured representations for fingerprint classification with recursive neural networks and support vector machines. In *Pattern Recognition* (2003), vol. 36, Elsevier, pp. 397–406.

[205] Yoon, S., Feng, J., and Jain, A. Fingerprint alteration. In *Proceedings of the 95th International Educational Conference of the International Association for Identification* (2010), pp. 11–17.

[206] Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. How transferable are features in deep neural networks? In *Advances in neural information processing systems* (2014), pp. 3320–3328.

[207] Zhang, D., Guo, Z., Lu, G., Zhang, L., Liu, Y., and Zuo, W. Online joint palmprint and palmvein verification. In *Expert Systems with Applications* (2011), vol. 38, Elsevier, pp. 2621–2631.

[208] Zhang, Z., Yan, J., Liu, S., Lei, Z., Yi, D., and Li, S. Z. A face antispoofing database with diverse attacks. In *ICB* (2012), IEEE, pp. 26–31.

[209] Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. Face recognition: A literature survey. In *ACM Comput. Surv.* (2003), vol. 35, ACM, pp. 399–458.

[210] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. Object detectors emerge in deep scene cnns. In *arXiv preprint arXiv:1412.6856* (2014).

[211] Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., and Oliva, A. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems* (2014), pp. 487–495.

[212] Zhu, J., and Hastie, T. Classification of gene microarrays by penalized logistic regression. In *Biostatistics* (2004), vol. 5, Oxford University Press, pp. 427–443.

[213] Zhu, J., and Hastie, T. Kernel logistic regression and the import vector machine. In *Journal of Computational and Graphical Statistics* (2005), vol. 14, Taylor & Francis, pp. 185–205.

[214] Zhu, Z., Luo, P., Wang, X., and Tang, X. Deep learning identity-preserving face space. In *Proceedings of the IEEE International Conference on Computer Vision* (2013), pp. 113–120.